PROGRAMAREA IN .NET - PARTEA 1 - INCEPUTURILE	2
PROGRAMAREA IN .NET - PARTEA A 2-A - CREEREA BAZEI DE DATE	3
PROGRAMAREA IN .NET - PARTEA A 3-A - CODUL PENTRU OBIECTE	8
PROGRAMAREA IN .NET - PARTEA A 4-A : .NET PROGRAMMING	14
PROGRAMAREA IN .NET - PARTEA A 5A -EDITAREA DE OBIECTE	25
PROGRAMAREA IN .NET - PARTEA A 6-A - SETUP PROGRAM	31
PROGRAMAREA IN .NET - PARTEA A 7-A : ASP.NET APPLICATION	35
PROGRAMAREA IN .NET - PARTEA A 8-A - EDITAREA IN ASP.NET	44
PROGRAMAREA IN .NET - PARTEA A 9-A -SITE MAP SI LOCALIZATION	57
PROGRAMAREA IN .NET - PARTEA A 10-A: UN PROIECT DOS	65
PROGRAMAREA IN .NET - PARTEA A 11-A TESTE AUTOMATE CU NUNIT	66
PROGRAMAREA IN .NET - PARTEA A 12-A DOCUMENTAREA - SCRIEREA DE FIȘIERE HELP.	74
PROGRAMAREA IN .NET - PARTEA A 13-A - LOG-AREA OPERAȚIILOR CU LOG4NET	82
PROGRAMAREA IN .NET - PARTEA A 14-A SALVARE XML SI EXECUTARE DE OPERAT ASINCRONE IN WINDOWS FORMS	ГН 85
PROGRAMAREA IN .NET - PARTEA A 15-A OPERATII ASINCRONE IN ASP.NET SI AJA	X 90
PROGRAMAREA IN .NET - PARTEA A 16-A RAPOARTE IN ASP.NET	99

## PROGRAMAREA IN .NET - PARTEA A 17-A RAPOARTE IN ASP.NET –WINDOWS FORMS 114

## **Programarea in .NET - partea 1 - Inceputurile**

Acesta este un prim articol dintr-o serie care vrea sa prezinte programarea in .NET( Windows si Internet) pe intelesul incepatorilor.

Ca sa puteti urma acest mini-curs, trebuie sa aveti cunostinte minime de programare.Nu o sa va fac teoria variabilelor, nici nu o sa va tin un curs de SQL si nici despre programarea 3-tier. Exemplele date vor fi facute in C# -dar ele se pot trece la fel de bine in VB.NET sau orice alt limbaj pe care .NET il suporta.Pentru mai multe limbaje, accesati adresa : <u>http://www.dotnetpowered.com/languages.aspx</u>

Ceea ce o sa ne intereseze in acest tutorial este partea de programare a lui NET.

Codul final este la adresa

http://serviciipeweb.ro/iafblog/content/binary/book2.zip

De ce aveti nevoie:

1. Un calculator cu Windows instalat( de preferinta, orice de la XP in sus)

2. IIS pentru proiectul Internet . Vedeti daca exista in Control Panel => Administrative Tools => Internet Information Services (IIS) Manager.

Daca nu, duceti-va la Control Panel => Add Or Remove programs => add / remove Windows components si il gasiti acolo

3. MSDN 2006 May - pentru tutoriale si exemple - free

:http://www.microsoft.com/downloads/details.aspx?FamilyID=373930CB-A3D7-4EA5-B421-DD6818DC7C41&displaylang=en

4.SQL Server Express - free : http://msdn.microsoft.com/vstudio/express/sql/download/

4. Visual Studio Web Development Express free - pentru proiecte internet -

http://msdn.microsoft.com/vstudio/express/vwd/)

5. Visual C# Express free - pentru proiecte windows -

http://msdn.microsoft.com/vstudio/express/visualcsharp/)

6. Optional : Report Viewer Control in Visual Studio 2005 free - pentru raportari locale - http://www.gotreportviewer.com/

Dupa ce ati downloadat si instalat aceste programe, prima aplicatie pe care o sa o facem este un proiect cu baze de date prin care o sa indexam cartile din biblioteca proprie.

Proiectul o sa fie schematic, doar pentru demonstrarea programarii Windows si Internet cu.NET. Elementele principale ale proiectului or sa fie Cartea, Autorul si Editura

Atributele principale ale Cartii sunt : Titlu Data Aparitiei ISBN Editura ( presupunem ca 1 carte nu este editata de mai multe edituri)

#### Ignat Andrei

Atributele principale ale Autorului sunt : Nume Prenume

Atributele principale ale Editurii sunt: Nume SiteWeb

O carte poate avea mai multi autori, iar un autor poate aparea pe mai multe carti ( legatura multi la multi)

O carte poate avea o singura editura , dar o editura poate publica mai multe carti

Ce se cere :

- forme de introducere a celor 3 obiecte
- cautare in baza de date dupa : Editura, Carte, Autor

- afisarea obiectelor in functie de relatiile dintre ele ( de exemplu, daca se selecteaza un autor, sa se afiseze toate cartile scrise de el )

Pina atunci, astept intrebarile voastre la adresa :contact@serviciipeweb.ro Lecturi Recomandate

Charles Petzold, DotNetBookZero – pentru incepatorii in .NET si nu numai <u>http://www.charlespetzold.com/dotnet/</u>

How to be a Programmer: A Short, Comprehensive, and Personal Summary

http://samizdat.mines.edu/howto/HowToBeAProgrammer.html

## programarea in .NET - partea a 2-a - creerea bazei de date

Conventie : notarile si codul si comentariile or sa fie in engleza. Nu de alta -dar e limba internationala a programarii.

Bun - acum ca ne-am lamurit ce vrem sa facem, hai sa concepem Baza de date. Aceasta o sa fie, pentru Windows forms, Access - iar pentru ASP.NET - SQL Express. De ce aceasta alegere ? Simplu :

Pentru aplicatia Windows forms - aleg ceva care sa fie usor de facut deployment-ul Pentru aplicatia ASP.NET - trebuie sa fie ceva care sa mearga repede - deci un SQL Server se impune

Cum o sa facem sa generam cod pentru oricare din acestea doua, o sa vedeti.

Structura Proiectului:

O sa cream proiectul in C:\Book

In acest director vom crea BookWin.sln ( solutia care va tine proeictul windows si proiectul Consola dos)

si

BookWeb.sln (solutia care va contine proiectul Web -si aplicatia SmartClient)

In acest director vom avea: BookData( fisierele mdb, scripturile de creeare sql express) BookObjects( proiectul de conectare la BD si de obiecte) BookWin( proiectul de Windows) BookDos ( proiectul de DOS) BookWeb(proiectul Web) BookDeployWeb(proiectul de deployment Web) BookDeployWin(proiectul de deployment Windows) BookDeploySmartClient(unde vom face deployment-ul SmartClient) BookTest Bun - acum haideti sa facem primul pas - creearea bazei de date Access.

Haideti sa creeam impreuna MDB-ul.

Pornim Acces - cream o noua baza de date - o salvam in BookData. O sa va arat cum se creeaza prima tabela:

Cind sunteti pe tabul "Tables" apasati pe "New Table"

🜆 book : Database	e (Access 2000 file format)	
🚰 Open 🕍 Design	n 🖪 New 🗙 🕒 🖆 📰 🏢	
Objects	Create table in Design view	
🔲 Tables	🚰 Create table by using wizard	
guenes	Create table by entering data	
📴 Forms	Author Author_Book	
Reports	🔲 Book	
🗎 Pages	🔲 Publisher	
📿 Macros		
💸 Modules		
Groups		
😹 Favorites		





incepeti sa introd	ок uceti valorile prezent	Cancel	Ş	000
Author : Table				
Field Name	Data Type			5
DAuthor	AutoNumber			
FirstNameAuthor	Text			
LastNameAuthor	Text			
-				
General Leaders				
Field Size	Long Integer			
New Values	Increment			
Format				
Caption				
Indexed	Yes (No Duplicates)			
Smart Tags				

Pentru a face IDAuthor Primary Key - dati click dreapta pe coloana de dinainte de IDAuthor si click pe "Primary Key"

Author : Table		
Field Name	Data Type	
IDAuthor Driver Key	AutoNumber	
Primary <u>K</u> ey	Text	
6 Cu <u>t</u>		
Copy		_
B. Paste		
Insert Rows		+
Delete Rows		
Build		-
Properties		
fel se creeaza si ce	lelalte tabele	+
fel se creeaza si ce um ar trebui sa le le	lelalte tabele. egam intre ele	+
fel se creeaza si ce um ar trebui sa le k ntru aceasta, accesa	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships
fel se creeaza si ce um ar trebui sa le k atru aceasta, accesa augati tabelele	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships
fel se creeaza si ce um ar trebui sa le k atru aceasta , accesa augati tabelele	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships
fel se creeaza si ce um ar trebui sa le k atru aceasta , accesa augati tabelele ow Table	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships
fel se creeaza si ce um ar trebui sa le k atru aceasta , accesa augati tabele le ow Table	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships ?×
fel se creeaza si ce um ar trebui sa le k atru aceasta , accesa augati tabelele ow Table	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships ?×
fel se creeaza si ce um ar trebui sa le k atru aceasta , accesa augati tabele le <b>ow Table</b> Tables Queries Both	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships ?×
fel se creeaza si ce um ar trebui sa le k atru aceasta , accesa augati tabele le <b>ow Table</b> <sup>Tables</sup> Queries Both	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships ?× Add 
fel se creeaza si ce um ar trebui sa le k atru aceasta , accesa augati tabelele <b>ow Table</b> <sup>Tables</sup> Queries Both Author Author Book	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships ?× Add Close
fel se creeaza si ce um ar trebui sa le k atru aceasta , accesa augati tabelele <b>ow Table</b> Tables Queries Both Author Author Book Publisher	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships ?× Add Close
fel se creeaza si ce um ar trebui sa le k atru aceasta , accesa augati tabele le <b>ow Table</b> Tables Queries Both Author Author Book Publisher	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships ?× Add Close
fel se creeaza si ce um ar trebui sa le k atru aceasta , accesa augati tabelele <b>ow Table</b> Tables Queries Both Author Author Book Publisher	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships ?× Add Close
fel se creeaza si ce um ar trebui sa le k atru aceasta , accesa augati tabelele <b>ow Table</b> Tables Queries Both Author Author Book Publisher	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships ?× Add Close
fel se creeaza si ce um ar trebui sa le k atru aceasta , accesa augati tabele le <b>ow Table</b> <sup>Tables</sup> Queries Both Author Author Book Publisher	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships ?× Add Close
fel se creeaza si ce um ar trebui sa le k atru aceasta , accesa augati tabele le ow Table Tables Queries Both Author Author Book Publisher	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships ?× Add Close
fel se creeaza si ce um ar trebui sa le k atru aceasta , accesa augati tabelele ow Table Tables Queries Both Author Author Book Publisher	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships ?X Add Close
fel se creeaza si ce um ar trebui sa le k atru aceasta , accesa augati tabele le <b>ow Table</b> Tables Queries Both Author Author Book Publisher	lelalte tabele. egam intre ele. ati Tools=> Relat	ionships Add Close

Tutorial programare in .NET	Ignat Andrei
🖩 🖥 Relationships	
Author FirstNameAuthor LastNameAuthor UBbook DBook DBook DAuthor	1 Publisher NamePublisher SitePublisher
Aveti grija sa selectati "referential entigrity" - cascade delete si update	
Edit Relationships	
IDAuthor       IDAuthor       OK         IDAuthor       IDAuthor       IDAuthor         IDAuthor       IDAuthor       IDAuthor	
Enforce Referential Integrity	
Cascade Update Related Fields	
Relationship Type: One-To-Many	

Acum avem Baza de date.Il puteti downloada de aici: book.mdb

Haideti sa concepem obiectele. Acestea or sa fie ca in prima prezentare:

<a

hre f="http://serviciipeweb.ro/iafblog/2006/08/10/Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx">Programarea+In+NET+Partea+1.aspx

In a treia parte o sa scriem cod pentru accesul la Baza de date, precum si obiectele principale pentru acest proiect.

•

## programarea in .NET - partea a 3-a - codul pentru obiecte

Acum la creearea obiectelor

Fiecare obiect trebuie sa aiba proprietati care corespund cimpurilor din Baza de date si, pentru usurinta, alte proprietati/metode

O sa scriu codul pentru Publisher si o sa las pe celelalte ca un exercitiu pentru dvoastra. Pentru a salva/sterge/creea un obiect de tip Publisher o sa implementam corespondentele metode corresponding method update / delete / insert.

Faptul ca un publisher trebuie sa aiba un nume unic o sa il scriem mai tirziu.

Stim deja ca trebuie sa avem doua baze de date, asa incit o sa avem doua stringuri de conexiune - pentru Access si pentru SQL Server)

Pentru a usura scrierea stringurilor de conexiune, va rog sa consultati www.connectionstrings.com

PEntru a vedea toate inregistrarile din tabela publisher, trebuie sa stocam undeva multimea lor. Aceasta multime va fi o clasa numita ColPublisher. Ea va contine o metoda numita Load care va incarca Publisher-ul din baza de date si le va stoca intr-o colectie.

Acum, cind stim ce avem de facut in continuare, hai sa scriem ceva cod

Start Visual C# Express( daca e prima oara cind il lansati, va va intreba setarile -puneti pe cele de C#)si creati un nou proiect numit Books

Application Console Application Starter Kit Collecti	? ×
Adates: sual Studio installed templates Windows Application Y Templates Starter Kit Starter Kit Starte	
Windows       Console       Empty Project       Screen Saver       Movie         Application       Application       Starter Kit       Collecti         y Templates       Console       Collecti	
Windows Cass Library Console Empty Project Screen Saver Movie Application Starter Kit Collecti	
Windows Class Library Console Empty Project Screen Saver Movie Application Starter Kit Collecti y Templates	
Application Starter Kit Collecti <b>y Templates</b>	
y Templates	τι Γ
earch Online emplates	
oject for creating a C# class library (.dll)	
: BookObjects	
	OK Cancel
ati in C:\book icei, the System.Data.dll este listat in referinte. Daca nu, va rog sa ati va rog si o referinta la System.Configuration. dreanta pe Solution Explorer si Add reference laOleDBConnectio	n il adaugati. Oricum,
dreapta pe solution Explorer si Add reference aoleo beofinectio	in ca mar jos .
umiti Class1.cs dinproprietati( click pe fisier in Solution Explorer her.cs	r si apasati F4) in
raspundeti "yes" la urmatoarea intrebare, numele clasei va fi schim her – si faceti-o public class	nbat din Class1 in
o sa scriem proprietatile de baza pentru un Publisher ·	
System;	
System.Collections.Generic;	
System.Text;	
pace BookObjects	

# Tutorial programare in .NET Ignat Andrei public class Publisher { #region Database properties private int m\_IDPublisher; public int IDPublisher { get { return m\_IDPublisher; } set { m\_IDPublisher = value; } } private string m\_Name; public string Name { get { return lame, } set { m Name = value; }

```
Tutorial programare in .NET
                                                                              Ignat Andrei
}
private string m Site;
public string Site
{
get
{
return m_Site;
}
set
{
m Site = value;
}
}
#endregion
}
}
Trebuie sa le incarcam aceste proprietati din Baza de date, asa incit o sa scriem o metoda Fill:
O sa o scriem astfel incit sa nu avem dependenta de BD:
      #region Database methods
public void FillObject(System.Data.IDataReader idr)
this.Name = idr["NamePublisher"].ToString();
this.Site = idr["SitePublisher"].ToString();
}
```

#endregion

ACum va trebui sa scriem codul de incarcare din BD a diverselor inregistrari. Asaugati o noua clasa (Project => Add Class) si denumiti-o ColPublisher.cs.

Din nou, faceti-o public class si haide sa scriem metoda care incarca datele din BD.

```
public void Load()
```

{

}

Cum ne dam seama ca ne trebuie cele doua conexiuni - si o metoda de a vedea pe care din ele le incarcam.

Asa incit vom incepe sa scriem cod spre a incarca conexiunile :

```
public static string ConnectionStringMDB
{
get
{
return System.Configuration.Configurat
                                           Manager.ConnectionStrings["MDB"];
}
}
public static string ConnectionStringSQLServer
{
get
return
       Configuration.ConfigurationManager.ConnectionStrings["SQLServer"];
System
}
Asa cum am tot spus, avem doua stringuri de conexiune.
A venit timpul sa stim cind incarcam una si cind cealalta in functia Load.
O sa adaug o noua clasa numita Settings si o sa pun acolo setarile comune
```

# Tutorial programare in .NET Ignat Andrei O sa adaug o enumerare care o sa imi spuna ce baza de date voi utiliza public enum DatabaseUsed { None, MDB, SQLServer } si o sa o citim din fisierul de configurare(App.config sau Web.Config) public static DatabaseUsed TheDatabase { get { return Enum.Parse(typeof(DatabaseUsed), System.Configuration.ConfigurationManager .AppSettings["DatabaseUsed"]; } } Puteti observa ca am pus configurarea in Appsetting. Putem stoca si stringurile de conexiune in acelasi loc - dar am vrut sa fim compatibili cu standardul Microsoft. Codul poate fi obtinut de aici Data viitoare o sa scriem cod pentru a incarca datele din BD si a le pune in colectie. De citit: Pentru NET best practices, puteti citi http://www.ssw.com.au/ssw/Standards/default.aspx Pentru construirea unui ORM (Object-relational mapping) cum facem aici, cititi articolul urmator de pe Wikipedia (http://en.wikipedia.org/wiki/Object-relational mapping) si puteti gasi o lista de ORM la adresa http://en.wikipedia.org/wiki/List of object-relational mapping software . Cred ca ar trebui sa cititi cel putin unul, de exemplu Nhibernate : http://www.hibernate.org/343.html Pentru SQL Server exista Data Access Application Blocks http://www.serviciipeweb.ro/iafblog Page 13

http://www.microsoft.com/downloads/details.aspx?familyid=f63d1f0a-9877-4a7b-88ec-0426b48df275&displaylang=en

Pentru a genera acelasi cod plecind de la tabele dintr-o baza de date, cititi Code Generation (<u>http://en.wikipedia.org/wiki/Code\_generation</u>) si poate CodeSmith (<u>http://www.codesmithtools.com/</u>)

## Programarea in .NET - partea a 4-a : .NET programming

Recapitulare: Acum avem baza de date Access in C:\Book\BookData – si dll-ul de incarcat datele se gaseste in folder-ul C:\Book\BookObjects.

Vom crea un proiect Windows Forms ca sa vedem datele care sunt in fisierul MDB.

Dati click cu dreapta pe solutie -si alegeti : Add => New Project => si selectati Windows Application. Scrieti numele BookWin. O forma noua va fi creata pentru voi.

Schimbati numele din Form1 in frmPublisherList.cs, dati dublu click pe fisier, apasati F4 ca sa aduceti fereastra de Properties in fata si schimbati Text din Form1 in List of Publishers.

Acum trebuie sa spunem proiectului windows sa utilizeze proiectul cu obiecte pe care l-am creat mai devreme.Dati click cu dreapta pe proiectul BookWin in Solution Explorer, click "Add reference" si dati click pe tab-ul Projects in urmatoarea fereastra.Dublu click pe BookObjects project.Compilati apasaind CTRL+SHIFT+B.

Trebuie sa gasim o cale sa copiem fisierul MDB in aceeasi locatie ca executabilul - in ideea ca asa o sa functioneze proiectul si daca aplicata va fi instalata de catre user si in alte locatii. (Nu orice user va fi de acord sa instaleze aplicatia in C:\Book ).

Solutia este sa facem copierea fisierului MDB de fiecare data dupa ce s-a facut un build - si operatia de fie automatizata.

Dati click dreapta pe proiectul BookWin din Solution Explorer, alegeti properties (sau le gasitit in meniul Project => Book Project Properties)

Pe tab-ul "build events" aveti "pre" si "post" evenimente de build :

Vom scrie acest cod in evenimentul de post:

copy \$(ProjectDir)..\BookData\\*.mdb \$(TargetDir)

Asta inseamna sa copieze toate fisierele MDB(\*.mdb) din C:\Book\BookData (obtinut din folder-ul proiectului (C:\Book\BookObjects\), apoi un folder mai sus (.. C:\Book\), apoi catre BookData (C:\Book\BookData)) in TargetDir (adica acolo unde se gaseste executabilul - nu conteaza daca suntem pe debug sau release.)

Salvati(CTRL+S) si compilati (CTRL + SHIFT + B)

Acum in C:\Book\BookObjects\bin\Release sau in C:\Book\BookObjects\bin\Debug trebuie sa fie inca o copie a fisierului MDB.

Pina aici e OK - haideti sa scriem stringul de conectare la MDB. Click cu dreaptape BookWin in Solution Explorer – click Add => New Item => si alegeti "Application Configuration File" ( nume default :App.config – nu il schimbati !) si sa scriem urmatoarele linii:

```
<?xml version="1.0" encoding="utf-8" ?>
```

<configuration>

<appSettings>

<add key="DatabaseUsed" value="MDB"/>

<!-- possible values : MDB, SQLServer-->

</appSettings>

<connectionStrings>

<add name="MDB" connectionString="Provider=Microsoft.Jet.OLEDB.4.0;Data Source=|DataDirectory|\book.mdb;User Id=admin;Password=;"/>

<!-- TODO : add for asp.net application the connection string with SQL Server-->

</connectionStrings>

</configuration>

#### Ignat Andrei

De ce am pus DataDirectory ? Daca aplicatia este de tip ClickOnce( mai multe date mai tirziu), atunci fisierele de date sunt puse in alta locatie decit executabilul.Acest datadirectory se va interpreta ca automat la folder-ul respectiv.

Acum este momentul sa incarcam datele - mai intii sa facem legatura la baza de date. Vom pune cod in fisierul settings.cs ca sa putem schimba conexiune dupa fisierul config la rulare :

#### public static DbConnection TheConnection

```
{
```

```
get
```

```
{
```

switch (TheDatabase)

```
{
```

case DatabaseUsed.MDB:

OleDbConnection oc = new OleDbConnection(ConnectionStringMDB);

return oc;

```
case DatabaseUsed.SQLServer:
```

SqlConnection sc=new SqlConnection(ConnectionStringSQLServer);

return sc;

default:

```
// Maybe throw an error that config file has not been initialized with
```

// the database type ?

return null

```
}
```

}

Dupa cum vedeti , intoarcem un DbConnection indiferent daca este ne conectam la Access sau SQL Server.

Si acum sa scriem cod generic de incarcat datele

public static IDataReader Load (string CommandLine, DbConnection dbcon)

```
{
```

if (!(dbcon.State == ConnectionState.Open))

```
dbcon.Open();
```

```
DbCommand dc = null;
```

```
switch (TheDatabase)
```

```
{
```

```
case DatabaseUsed.MDB:
```

```
dc = new OleDbCommand(CommandLine);
```

break;

```
case DatabaseUsed.SQLServer
```

```
dc = new SqlCommand(CommandLine);
```

break;

default:

 $//{\tt TODO}$  : throw specific error that database type does not properly have been initialized

```
}
```

break;

```
dc.CommandType = CommandType.Text;
```

dc.Connection = dbcon;

```
return dc.ExecuteReader();
```

Ignat Andrei

return null;

}

Sa incarcam datele in clasa colectie, ColPublisher :

using System;

using System.Collections.Generic;

using System.Text;

using System.Data;

using System.Data.Common;

namespace BookObjects

{

```
public class ColPublisher :
System.Collections.ObjectModel.KeyedCollection<string,Publisher>
```

{

protected override string GetKeyForItem(Publisher item)

return "K" + item.IDPublisher;

}

```
public void Load()
```

{

DbConnection db = Settings.TheConnection;

Ignat Andrei

using (db)

{

db.Open();

IDataReader ir = Settings.Load("select IDPublisher, NamePublisher, SitePublisher from Publisher", db);

while (ir.Read())

{

}

}

}

}

Publisher p = new Publisher();

p.FillObject(ir);

//TODO : add p into the collection

}			

Acum ne dam seama ca nu am scris ColPublisher ca si o colectie unde putem sa adaugam/stergem/identifica un Publisher. Din fericire, .NET are destule colectii ajutatoare.

Sunt 3 namespace-uri mari care contin colectii :System.Collections , System.Collections.ObjectModel si System.Collections.Specialized. Veti gasi mult mai mult pe Internet( de exemplu o implementare an implementation of a Set collection : <a href="http://www.codeproject.com/cshap/sets.asp">http://www.codeproject.com/cshap/sets.asp</a>)

VomallegepentruColPublisherclasaSystem.Collections.ObjectModel.KeyedCollection<string,Publisher>.0satrebuiasca sa scriem cum se obtine o cheie unica(identificator unic) pentru un specific Publisher -sice e o mai buna alegere decit ID ?

protected override string GetKeyForItem(Publisher item)

{

```
Ignat Andrei
```

return "K" + item.IDPublisher;

}

Acum putem scrie metoda Load:

using System;

using System.Collections.Generic;

using System.Text;

using System.Data;

using System.Data.Common;

```
namespace BookObjects
```

{

```
public class ColPublisher :
System.Collections.ObjectModel.KeyedCollection<string,Publisher>
```

```
{
```

protected override string GetKeyForItem(Publisher item)

```
{
```

return "K" + item.IDPublisher;

```
}
```

```
public void Load
```

{

DbConnection db = Settings.TheConnection;

using (db)

{

```
db.Open();
```

IDataReader ir = Settings.Load("select IDPublisher, NamePublisher, SitePublisher from Publisher", db);

Ignat Andrei

while (ir.Read())
{
Publisher p = new Publisher();
p.FillObject(ir);
this.Add(p);
}
}
}

Sa vedem cum o folosim din form:

Faceti dublu click pe frmPublisherList.cs si trageti un DataGridView pe forma

### Ignat Andrei

Toolbox	- # ×	frmPublisherList.cs [Design]	
± All Windows F	orms		
🗄 Common Cont	rols	List of Publishers	
🗄 Containers			have brinkering
🗉 Menus & Tool	bars		
🖻 Data			
Pointer			
🛃 DataSet			
🔁 DataGridView			
BindingSource	DataGridView Version 2.0.0.0 from Microsoft	Corporation	þ
± Components	.NET Component	7M	
± Printing	Displays rows and columns of d	ata in a grid you can	
🗄 Dialogs	customize,		
🗄 General			
There are no usab an item onto this l	le controls in this group. Drag text to add it to the toolbox.		
		MC.	

Sa configuram acum data source:

🔜 List of Publishers	
·	DataGrid¥iew Tasks
	Choose Data Source (none)
	Edit Columns
	Add Column
	🔽 Enable Adding
	🗹 Enable Editing
	🔽 Enable Deleting
0	Enable Column Reordering
	Dock in parent container

·.<V

Click pe "(none)" si selectati "add new data source"

In dialogul urmator alegeti "Object"

#### //trmPublisherList.cs [Design]\* |

🔛 List of Publishers	
9	DataGridView Tasks
	Choose Data Source (none)
	🎁 Add Project Data Source
	Click the 'Add Project Data Source' link to connect to data.

Ł

/here will t	he application g	et data from?	
Database	Web Service		
Database	WED DELVICE		

si apasati next

Acum mariti nodul BookObjects si alegeti "ColPublisher". Apasati Next si apoi Finish.

Pe forma a aparut un nou control numit colPublisherBindingSource - " si Grid-ul are deja coloanele definite.

Acum sa scriem codul pentru a incarca datele din Baza de date:

Dublu dick pe forma si o sa va gasiti editind evenimentul Form\_Load:

```
private void frmPublisherList Load(object sender, Event)
```

```
{
```

BookObjects.ColPublisher publishers = new BookObjects.ColPublisher();

publishers.Load();

```
colPublisherBindingSource.DataSource = publishers;
```

```
}
```

Sa setam solutia sa porneasca cu BookWin - " click dreapta pe BookWin - " si selectati "Set as startup project"

Acum apati F5 si asteptati sa vedeti rezultatele. Daca totul e OK, veti vedea forma fara nici un fel de date - si e foarte correct din cauza ca nu am introdus nici un fel de Publisher-i in Book.mdb

In urmatoarea lectie vom scrie cod ca sa inseram un nou publisher si sa il vedem in lista

Lecturi optionale:

CSLA : <u>http://www.lhotka.net/cslanet/</u> - " un framework bun pentru a se folosi pentru securitate, scalabilitate, binding si multe altele(si, mai ales, free si cu cod sursa).

## Programarea in .NET - partea a 5a -editarea de obiecte

Va trebui, asa cum am promis, sa facem adaugarea/ modificarea / stergerea de obiecte( pe scurt, C(R)UD – create, (read), update, delete). Luam un buton din Toolbox, il tragem pe forma, ii setam din proprietati ( apasati F4) numele la btnAdd si text la "&Add" ( &A e pentru ca, atunci cind apasam <ALT> + A, sa fie ca si cind dam click pe button)

Acum dati dublu click pe button – si veti intra in codul de click. Avem o problema : trebuie ca user-ul sa introduca numele Publisher-ului. O sa creeam o noua forma : Click dreapta pe BookWin, Add => Windows Form- si ii veti da denumirea de frmPublisherAdd.cs. Apasati F4 si la Text puneti : Add Publisher

Adaugati un Label( Name: lblName, Text: &Name) si un textbox ( Name: txtName)

Adaugam acum un Button de Add (Name : btnAdd, Text : &Add) si unul de Exit (Name : btnExit, Text : E&xit)

Codul de pe btnExit e cel mai usor ( dati dublu click pe buton)

private void btnExit\_Click(object sender, EventArgs e)

this.Close();//close the form

Codul de pe Button-ul de Add :

vate void btnAdd\_Click(object sender, EventArgs e)

BookObjects.Publisher p = new BookObjects.Publisher();

p.Name = txtName.Text;

p.Save();

this.Close();//close the form

Ignat Andrei

```
}
E clar ca trebuie sa scriem metoda de Save pe Publisher
Inapoi la clase : si acolo vom scrie metoda de save.
        public void Save()
         {
             string strSQL = " insert into Publisher(NamePublisher
             if (!string.IsNullOrEmpty(this.Site))
                 strSQL += ",SitePublisher ";
             strSQL += " ) Values (";
             strSQL += "'" + this.Name.Replace("'","''') + "'";//terminator for
string in SQL is ' - so replace with
             if (!string.IsNullOrEmpty(this.Site))
                        L += "," +"'"+ this.Site.Replace("'", "''")
string in SQL is ' - so replace with ''
                 strSQL += ", " +"" +
+"'";//terminator for
               rSOL
             Settings.ExecuteSQL(strSQL);
```

```
Tutorial programare in .NET
```

Ignat Andrei

}

In sfirsit, sa scriem codul pentru adaugare de pe forma de list:

E clar ca va trebui sa facem re incarcarea datelor – deci o sa luam codul de pe frmPublisherList Load si o sa il punem in o functie generica , RebindData()

private void btnAdd Click(object sender, EventArgs e)

{

}

{

frmPublisherAdd f = new frmPublisherAdd();

f.ShowDialog(this);

RebindData();

```
private void RebindData()
```

BookObjects.ColPublisher publishers = new BookObjects.ColPublisher();

publishers.Load();

colPublisherBindingSource.DataSource = publishers;

Sa verificam functionarea

Dati CTRL+ F5 , apasati Add - introduceti un nume - apasati Add - si verificati ca se vede in lista ceea ce ati introdus.

Sa facem acum stergerea.

Adaugati alt buton(Name = btnDelete, Text = &Delete) si sa scriem cod pentru delete

```
Tutorial programare in .NET
                                                                        Ignat Andrei
private void btnDelete Click(object sender, EventArgs e)
        {
            BookObjects.Publisher p = colPublisherBindingSource.Current as
BookObjects.Publisher;
            if (p != null)
            {
                //avert the user
                if (MessageBox.Show(this, "Delete " + p.Name,
                                                                 "Delete",
MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.No)
                    return;
                p.Delete();
                RebindData();
            }
        }
In sfirsit, sa scriem codul pentru update: iarasi buton, iarasi cod
private void btnUpdate Click(object sender, EventArgs e)
            BookObjects.Publisher p = colPublisherBindingSource.Current as
BookObjects.Publisher;
            if (p == null)
                return;
```

}

frmPublisherUpdate f = new frmPublisherUpdate(p);

f.ShowDialog(this);

RebindData();

Pentru asta, adaugam o alta forma, in care sa facem update .

Dar avem neaparata nevoie de un publisher pe care sa facem update.Vom modifica constructorul formei ca sa accepte ca parametru de intrare un publisher.

Ca sa punem valorile deja obtinute in text box-uri, avem 2 variante :

Fie codam de mina de doua ori ( ceva de genul txtName.Text= m\_Publisher.Name si , pe salvare, m\_Publisher.Name= txtName.Text),fie lucram cu DataBindings direct . Prefer acum, pentru rapiditate, a doua varianta :

public partial class frmPublisherUpdate : Form

{

{

{

private BookObjects.Publisher m Publisher;

public frmPublisherUpdate(BookObjects.Publisher pub)

m\_Publisher = pub;

InitializeComponent();

private void frmPublisherUpdate Load(object sender, EventArgs e)

txtName.DataBindings.Add("Text", m Publisher, "Name");

```
Tutorial programare in .NET
```

#### Ignat Andrei

```
txtSite.DataBindings.Add("Text", m Publisher, "Site");
        }
        private void btnSave Click(object sender, EventArgs e)
        {
            m_Publisher.Update();
           this.Close();
        }
       private void btnExit_Click(object sender, EventArgs
                                                              e)
        {
            this.Close();
        }
   }
Sa scriem si codul de salvare pe publisher :
               void Update()
        public
             tring strSQL = " update Publisher set ";
            strSQL += " NamePublisher = ";
            strSQL += "'" + this.Name.Replace("'", "''") + "'";
            strSQL += ",";
```

```
http://www.serviciipeweb.ro/iafblog
```

}

#### Ignat Andrei

```
strSQL += " SitePublisher = ";
```

```
strSQL += "'" + this.Site.Replace("'", "''") + "'";
```

```
strSQL += " where IDPublisher = " +
this.IDPublisher;
```

strSQL += " ";

Settings.ExecuteSQL(strSQL);

Ca tema de acasa, ramine sa faceti acelasi lucru pentru tabela Author.

Data viitoare o sa facem un mic refactoring de code... si o sa facem un program de setup pentru aplicatia Windows.

## **Programarea in .NET - partea a 6-a - setup program**

Setup-ul este proiectul prin care creeam un kit de instalare – presupunem ca vrem sa scriem un CD cu aplicatia(dll, exe, BD, alte setari).

Click dreapta pe BookWin, alegeti "properties" si click pe tab-ul de "Publish"

Acolo gasiti "Publishing location" - "valoare implicita fiind http://localhost/BookWin/

Daca nu aveti IIS, atunci puteti alege si o cale fizica, de exemplu : C:\Book\BookSetup\

Problema de rezolvat : printre fisierele care trebuie instalate se afla si fisierul mdb.

Daca aveti VS.NET Standard(sau ceva mai mult) aveti un proiect special care va ingaduie sa adaugati alte fisiere la proiectul de instalare. Pentru moment ne jucam cu ce avem. Asa ca faceti dick dreapta pe proiectul Bookwin, click add=>existing item si cautati book.mdb file din C:\Book\BookData.

Compilati proiectul (CTRL+Shift+B) si veti gasi acum book.mdb on intre fisierele aplicatiei – ca fisier de date.

Acum conditiile de instalare : Pentru ca facem instalarea de pe un CD, e mai bine sa includem si kitul de .NET 2.0 –ca sa facem download tot de pe CD(cind facem instalarea de pe Internet, e preferabil sa facem instalarea .NET 2.0 de la site-ul MS)

rerequisites	<u>? ×</u>
Create setup program to install prerequisite components	
Choose which prerequisites to install:	
🗹 🚰 .NET Framework 2.0	
🔲 🚰 Windows Installer 3.1	
🔲 🖾 SQL Server 2005 Express Edition	
Check Microsoft Update for more redistributable components	
Specify the install location for prerequisites	
O Download prerequisites from the component vendor's web site	
Download prerequisites from the same location as my application	
Download prerequisites from the following location:	
	Browse
OK	Cancel
	 Cancer

Update-urile automane nu sunt inca valabile, din cauza ca nu avem inca un WebSite. Dar putem modifica citeva dintre Options, ca "publisher name", "product name" si altele.

Acum sa apasam pe Publish Wizard :

Primul pas ne intreaba un sa creeam kit-ul de instalare (C:\Book\BookSetup\deja selectat), urmatorul sa ne intrebe suportul fizic (click "from cd -"rom or dvd rom").

Al treilea pas ne ingaduie sa setam unde va cauta aplicatia update-urile – dar, asa cum am spus nu avem inca un WebSite asa ca selectati "the application will not check for updates"

Si apasati finish!

Daca vreo eroare se intimpla spunind "can not find package", mergeti la Microsoft .NET FrameworkVersion2.0RedistributablePackage(x86)http://www.microsoft.com/downloads/details.aspx?FamilyID=0856eacb-4362-4b0d-8edd-aab15c5e04f5&displaylang=ensi downloadati kit-ul. Copiati-l in <C:\Program Files>\Microsoft VisualStudio 8\SDK\v2.0\Bootstrapper\Packages\DotNetFX

Pentru instmsia.exe , duceti-va la http://go.microsoft.com/fwlink/?LinkId=37285

Acum puteti scrie folder-ul C:\Book\BookSetup pe un CD si sa testati instalarea.

Daca nu vreti sa irositi un CD, puteti scrie folder-ul intr-un fisier .iso si sa il incarcati ca un CD.

O metoda de a creea un fisier ISO este Alex Feinman IsoRecorder

http://isorecorder.alexfeinman.com/isorecorder.htm

Faceti download ( eu am testat versiunea pentru XP SP2 ) si click dreapta pe folder-ul Book Setup

🚔 C:\Book	
File Edit View Favorites Tools Help	
🕞 Back 🝷 🕥 🖌 🏂 🔎 Search 🦷	
Address 🗀 C:\Book	
Name 🔺	
🚞 BookData	
BookObjects	
BookSetun	
C Open	O
C Explore	
e Search	
Sharing and Security	
🤄 📢 Compare To	
Compare	
Add to archive	
Add to "BookSetup.rar"	
Compress and email	
Create ISO impact file	
Create 150 image file	
Scan for Viruses	00
Send To	
Cut	
Сору	
Create Shortcut	
Delete	
Rename	
Properties	

In meniu vedeti "Create ISO image file " click si acceptati setarile existente.

Acum aveti un CD -si puteti incarca acest CD cu Virtual CD Control Panel.

Puteti testa acum programul de pe CD-ul incarcat.

Mai mult, daca vreti sa il testati pe alta versiune de Windows (pentru care aveti kit-ul de instalare) puteti face download la Virtual PC 2004 SP1 or VMWare si sa creati cu aceasta un nou Windows (repet : trebuie sa aveti CD-urile de instalare a Windows!).

Data viitoare o sa facem o aplicatie ASP.NET cu o baza de date SQL Server.

Lecturi recomandate:

ComparatieintreversiuniledeVS.NET2005:http://msdn.microsoft.com/vstudio/products/compare/default.aspx

VirtualPC2004SP1(free)http://www.microsoft.com/downloads/details.aspx?familyid=6d58729d-dfa8-40bf-afaf-20bcb7f01cd1&displaylang=en

Probleme de instalare http://forums.microsoft.com/MSDN/ShowPost.aspx?PostID=13937&SiteID=1

Virtual CD Control Panel - "<u>http://download.microsoft.com/download/7/b/6/7b6abd84-7841-</u> 4978-96f5-bd58df02efa2/winxpvirtualcdcontrolpanel\_21.exe

Alex Feinman - "Make ISO http://isorecorder.alexfeinman.com/isorecorder.htm

# Programarea in .NET - partea a 7-a : ASP.NET application

Acum e cazul sa facem o aplicatie ASP.NET. Presupun ca ati instalat deja Internet Information Services (daca il aveti, gasiti un shortcut in folder-ul Administrative tools) .Daca nu , va rog sa il instalati de pe CDul cu Windows si sa rulati aspnet\_regiis.exe pe care o sa il gasiti in <WindowsPath> \Microsoft.NET\Framework\<latest version> (my path is C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727)

Instalati SQL Server Express(<Address>) si WebDeveloper(<Adress>) –acesta fara instalare de SQL pentru ca deja ati facut-o.

Acum e timpul sa importam datele noastre in baza de date SQL Server. Porniti SQL Server Management Studio si o sa gasiti citeva baze de date deja instalate.Creati una noua ( nume : Book) si importati mdb-ul in aceasta noua baza de date. Click dreapta pe baza de date Book => Tasks= > Import data .

In primul ecran alegeti ca sursa de date Microsoft Access

Tutorial pr	ogramare	in	.NET
-------------	----------	----	------

lσr	hat .	∆nd	rei
ıgı	iat i	- IU	

📃 SQL Server Import	and Export Wizard	
Choose a Data S Select the source fr	Source rom which to copy data.	
<u>D</u> ata source:	🔦 Microsoft Access	•
To connect, select a advanced options.	database and provide a user name and password. You	may need to specify
F <u>i</u> le name:	C:\Book\BookData\book.mdb	Browse
<u>U</u> ser name:		
<u>P</u> assword:		
	<u>A</u> dvanced	
<u>H</u> elp	< <u>B</u> ack <u>N</u> ext > <u>Fi</u> nit	sh>>>  Cancel
	alle.	

Urmatorul ecran e configurat OK – dupa cum vedeti utilizeaza windows authentication.

Alegeti apoi "Copy data from one or more table or views" si click "select all" in urmatorul ecran.

Puteti acum apasa "Finish" si sa asteptati terminarea operatiei.

Acum citeva modificari ale structurii se impun pentru a nu avea probleme de concurenta la creearea simultana a doi Publisher-i :
Alegeti Book=> Tables in SQL Server Management Studio. Click dreapta tabela Publisher si apasati "modify".

Vom face IDPublisher un "auto number" ca in Access. In SQL Server, "auto number" se numeste "identity".

Column Name       Data Type       Allow Nulls         DPrublisher       int       □         NamePublisher       nvarchar(S0)       ✓         StePublisher       nvarchar(S0)       ✓         StePublisher       nvarchar(S0)       ✓         StePublisher       nvarchar(S0)       ✓         Image: StePublisher       nvarchar(S0)       ✓         Image: StePublisher       nvarchar(S0)       ✓         Image: StePublisher       No          Image: StePublisher       IDPublisher       No         Image: StePublisher       No          Image: StePublisher       No          Allow Nulls       Image: StePublisher       No         Default Value or Binding       Image: StePublisher          Image: StePublished       No          Table Designer       int          Condensed Data Type       int          Description           Description        No         Description        No         Identity Increment       1          Identity Seed       1          Indexable       No </th <th>able - dbo.Publisher*</th> <th>Summary</th> <th></th> <th></th>	able - dbo.Publisher*	Summary		
IDPublisher       int         NamePublisher       nvarchar(50)       Image: StePublisher         StePublisher       nvarchar(50)       Image: StePublisher         StePublisher       nvarchar(50)       Image: StePublisher         Jume       Image: StePublisher       Image: StePublisher         Jume       Image: StePublisher       Image: StePublisher         Jume       Jume       Image: StePublisher         Jume       Jume       Image: StePublisher         Allow Nulls       Image: StePublisher       No         Data Type       int       Image: StePublisher         Jume            Data Type       int       Image: StePublished         Jume	Column Name	Data Type	Allow Nulls	
NamePublisher       nvarchar(50)       ✓         SitePublisher       nvarchar(50)       ✓         SitePublisher       nvarchar(50)       ✓         SitePublisher       I       I         SitePublisher       I       I         SitePublisher       I       I         SitePublisher       IDPublisher       IDPublisher         Allow Nulls       No       No         Default Value or Binding       int       IDPublisher         IDefault Value or Binding	IDPublisher	int		
StePublisher     nvarchar(S0)     ✓       Jumn Properties     □       Jumenty Specification     □	NamePublisher	nvarchar(50)	~	
skum Properties	SitePublisher	nvarchar(50)		
olumn Properties				
Image: Stress			-	
Juli Ceneral         (General)         (General)         (Name)       IDPublisher         Allow Nulls       No         Deta Type       int         Default Value or Binding       int         Table Designer       Collation         Collation <database default="">         Condensed Data Type       int         Description       int         Detarype       int         Identity Speci</database>				
olum Properties           ↓ ↓ □         (General)         (General)         (Name)       IDPublisher         Allow Nulls       No         Data Type       int         Default Value or Binding       int         Table Designer       Collation         Collation          Condensed Data Type       int         Description          Detauts Specification          Condensed Data Type       int         Description          Detautished       No         Pathext Specification       No         Pathext Specification          Identity Specification          Identity Specification       No         Has Non-SQL Server Subscriber       No         Identity Specification       Yes         Marge-published       No         No       No         Replicated       No         Replicated       No         RowGuid       No				
Image: State Sta				
Image: State Sta				
olumn Properties				
Image: Second Secon				
Image: Image				
Ceneral         (Name)       IDPublisher         Allow Nulls       No         Data Type       int         Default Value or Binding       int         Table Designer       Collation         Collation <database default="">         Condensed Data Type       int         Description       int         Description          Deterministic       Yes         DTS-published       No         Identity Specification       No         Identity Specification       Yes         Identity Specification       Yes         Identity Specification       Yes         Identity Specification       No         Identity Specification       No         Identity Specification       No         No       Identity Increment         Identity Seed       1         Indexable       Yes         Merge-published       No         No       No         Replication       No         Replication       No         Replicated       No         Replicated       No         Replicated       No         Size       4</database>				
Centeral y         IDPublisher           Allow Nulls         No           Data Type         int           Default Value or Binding            Table Designer            Collation <database default="">           Condensed Data Type         int           Description         int           Description            Detruitsic         Yes           DTS-published         No           Identity Specification         Yes           Identity Specification         Yes           Identity Seed         1           Identity Seed         1           Identity Seed         1           Identity Seed         No           Image-published         No           Not For Replication         No           Replicated         No           Replicated         No           Replicated         No  </database>				
Value     Default       Allow Nulls     No       Data Type     int       Default Value or Binding        Table Designer        Collation <database default="">       Condensed Data Type     int       Description        Description        Detruinistic     Yes       Toble Description     No       Full-text Specification     No       Has Non-SQL Server Subscriber     No       Identity Specification     Yes       Identity Specification     Yes       Identity Specification     1       Indexable     Yes       Identity Seed     1       Indexable     No       Not For Replication     No       Replicated     No       Replicated     No</database>	(General)			TOD. Lister.
Allow Nuls         No           Data Type         int           Default Value or Binding         int           Collation            Collation            Conputed Column Specification         int           Description         int           Description         int           Deterministic         Yes           DTS-published         No           Has Non-SQL Server Subscriber         No           Identity Specification         Yes           Merge-published         No           Not For Replication         No           Replicated         No           Replicated         No           Size         4	(Ivame)			
Default Value or Binding         Int           Image: Provide Column Specification            Condensed Data Type         int           Description            Deterministic         Yes           DTS-published         No           Identity Specification         Yes           Identity Specification         No           Merge-published         No           Not For Replication         No           Replicated         No           Replicated         No           Replicated         No	Allow Nulls			NO
Table Designer         Collation <database default="">         Conjusted Column Specification       int         Condensed Data Type       int         Description          Deterministic       Yes         DTS-published       No         Has Non-SQL Server Subscriber       No         Identity Specification       Yes         Identity Seed       1         Indexable       Yes         Merge-published       No         Not For Replication       No         Replicated       No         Replicated       No         Replicated       No         Size       4</database>	Data Type	-		
Collation <database default="">         Collation       <database default="">         Computed Column Specification       int         Condensed Data Type       int         Description          Deterministic       Yes         DTS-published       No         Full-text Specification       No         Has Non-SQL Server Subscriber       No         Identity Specification       Yes         Identity Specification       No         Not For Replication       No         Not For Replication       No         Replicated       No         RowGuid       No         Size       4</database></database>	Derault value or binding	,		
Collation <database derault="">         Computed Column Specification       int         Condensed Data Type       int         Description          Deterministic       Yes         DTS-published       No         Full-text Specification       No         Has Non-SQL Server Subscriber       No         Identity Specification       Yes         Identity Specification       Yes         Identity Increment       1         Identity Seed       1         Indexable       Yes         Merge-published       No         Not For Replication       No         Replicated       No         RowGuid       No</database>	Table Designer			
Computed Column Specification       int         Condensed Data Type       int         Description       Pescription         Deterministic       Yes         DTS-published       No         Identity Specification       No         Has Non-SQL Server Subscriber       No         Identity Specification       Yes         Identity Specification       Yes         Identity Increment       1         Identity Seed       1         Indexable       Yes         Merge-published       No         Not For Replication       No         Replicated       No         Replicated       No				<database derault=""></database>
Condensed Data Type       int         Description       Ves         Dts-published       No         Tul-text Specification       No         Has Non-SQL Server Subscriber       No         Identity Specification       Yes         Identity Seed       1         Indexable       Yes         Merge-published       No         Not For Replication       No         Replicated       No         Replicated       No         Size       4		irication		· .
Description         Yes           Deterministic         Yes           DTS-published         No           Intext Specification         No           Has Non-SQL Server Subscriber         No           Identity Specification         Yes           Identity Specification         Yes           Identity Specification         Yes           Identity Increment         1           Identity Seed         1           Indexable         Yes           Merge-published         No           Not For Replication         No           Replicated         No           RowGuid         No           Size         4	Condensed Data Type			int
Deterministic       Yes         DTS-published       No         Image: Pull-text Specification       No         Has Non-SQL Server Subscriber       No         Identity Specification       Yes         Identity Increment       1         Identity Seed       1         Indexable       Yes         Merge-published       No         Not For Replication       No         Replicated       No         Replicated       No         Size       4	Description			
D1S-published       No         Full-text Specification       No         Has Non-SQL Server Subscriber       No         Identity Specification       Yes         Identity Increment       1         Identity Seed       1         Indexable       Yes         Merge-published       No         Not For Replication       No         Replicated       No         Replicated       No         Identity Seed       No         Indexable       Yes         Merge-published       No         Not For Replication       No         Replicated       No         RowGuid       No         Size       4	Deterministic			Yes
Heal-text Specification       No         Has Non-SQL Server Subscriber       No         Identity Specification       Yes         Identity Increment       1         Identity Seed       1         Indexable       Yes         Merge-published       No         Not For Replication       No         Replicated       No         Size       4	DTS-published			No
Has Non-SQL Server Subscriber       No         Identity Specification       Yes         (Is Identity)       Yes         Identity Increment       1         Identity Seed       1         Indexable       Yes         Merge-published       No         Not For Replication       No         Replicated       No         Size       4	<ul> <li>Full-text Specification</li> </ul>			No
Identity Specification     Yes       (Is Identity)     Yes       Identity Increment     1       Identity Seed     1       Indexable     Yes       Merge-published     No       Not For Replication     No       Replicated     No       Size     4	Has Non-SQL Server Su	ibscriber		No
Yes       Identity Increment     1       Identity Seed     1       Indexable     Yes       Merge-published     No       Not For Replication     No       Replicated     No       Size     4	Identity Specification			Yes
Identity Increment     1       Identity Seed     1       Indexable     Yes       Merge-published     No       Not For Replication     No       Replicated     No       Size     4	(Is Identity)			Yes
Identity Seed1IndexableYesMerge-publishedNoNot For ReplicationNoReplicatedNoRowGuidNoSize4	Identity Increment			1
Indexable     Yes       Merge-published     No       Not For Replication     No       Replicated     No       RowGuid     No       Size     4	Identity Seed			1
Merge-published     No       Not For Replication     No       Replicated     No       RowGuid     No       Size     4	Indexable			Yes
Not For Replication         No           Replicated         No           RowGuid         No           Size         4	Merge-published			No
Replicated         No           RowGuid         No           Size         4	Not For Replication			No
RowGuid     No       Size     4	Replicated			No
Size 4	RowGuid			No
	Size			4

Imediat ce modificati proprietatea (ls Identity), veti vedea "identity increment" si "seed " fiind modificate in 1 si 1( adica creste cu 1(increment) pornind de la 1(seed)).

Salvati tabela.

Urmatorul pas este sa creeam un folder in care sa creeam program-ul internet.

Duceti-vala C:\Book si creati un nou folder , numit BookWeb

Acum din folder-ul Administrative Tools faceti dublu click pe Internet Information Services si mariti (local computer)=> WebSites=> Default Web Site.Click dreapta pe Default Web Site si alegeti New => Virtual Directory. Apasati pe next. In "Alias" introduceti numele BookWeb. Pentru folder alegeti C:\Book\BookWeb .Apasati de doua ori "Finish" si suntem gata sa incepem.

Acum suntem gata sa incepem prima noastra pagina in ASP.NET.

Ca si in proiectul Windows vom avea o pagina de lista si paginile de creere/modificare/stergere pentru Publisher.Ca sa facem ca site-ul sa aiba aceeasi infatisare, vom crea o pagina "master" numita -"Book.master".

Deschideti Microsoft Visual Web Developer 2005 Express Edition si alegeti "Open Web Site" ca in imagine:



Alegeti tabul "Local IIS" si "BookWeb" ca "virtual directory"

In "Solution Explorer" click dreapta pe <u>http://localhost/BookWeb</u> si alegeti "Add new Item " si alegeti Master Page si numiti "Book.master".Nu uitati sa selectati "Place code in separate file"

dd New Item	- http://localhost/B	lookWeb/		?
emplates:				
Visual Stud	lio installed templat	es		_
Web Forr HTML Page Style She SQL Data SQL Data SQL Data SQL Data Mobile We Mobile We Source F My Templa	n ge let base eb Configuration File =ile <b>ltes</b> nline Templates	Master Page Web Service Clobal Application Class Text File DataSet Mobile Web Form Script File	<ul> <li>B: Web User Control</li> <li>Class</li> <li>Web Configuration File</li> <li>Resource File</li> <li>Generic Handler</li> <li>VBScript File</li> <li>Mobile Web User Control</li> <li>Skin File</li> </ul>	
A Master Page	for Web Applications			
ame:	Book.master			
anguage:	Visual C#	▼ Place coo	<b>le in separate file</b> aster page	
			Add	Cancel
		ich		
n face inte	erfata grafica a n	naster page mai degraba sir	mpla, lasind cititorului sarcina de	ао
			rindul do que fiind formet dintr o e	

Codul este urmatorul:

celula

<title>Untitled Page</title>

```
Tutorial programare in .NET
                                                  Ignat Andrei
</head>
<body>
  <form id="form1" runat="server">
     <div>
        Book application
           Right menu
              <asp:ContentPlaceHolder ID="ContentPlaceHolder1"
runat="server">
                      ContentPlaceHolder>
</body>
</html>
```

Acum sa creeam pagina de listare de publisher. Adaugati un nou item -" Web Form(frmPublisherList.aspx) -" si verificati ca ati check-uit amindoua check-urile.Urmatorea fereasta va arata BookMaster pe care trebuie sa il selectati si apoi sa apasati OK.Click dreapta pe fisier si alegeti "Set as start page"

In asp:content trageti un gridview (il gasiti in Toolbox=> Data).

Acest grid trebuie umplut cu date – asa ca trebuie sa ne conectam la obiectele create(la fel ca la aplicatia Windows). Aceasta e problema –cum putem sa ne conectam fie la versiunea de Release sau debug a proiectului daca nu il avem in solutie, asa cum era la Windows Forms ?Solutia este indirectia(destul de intilnita in programare) : facem un nou folder, bookdll si o sa copiem acolo obiectele, indiferent daca e debug sau release. Din acest nou folder o sa isi ia obiecteleaplicatia Web.

Creati folder-ul C:\Book\BookDll , mergeti inapoi la solutia Book.sln si la Build puneti aceasta linie in "Post build command line":

copy \$(TargetDir)\*.\* \$(ProjectDir)..\BookDll /Y

Compilati proiectul .O sa gasiti in folder-ul C:\Book\BookDll fisierul BookObjects.dll

Adaugati referinta la BookObjects.dll aplicatiei BookWeb ( ca de obiœi: dick dreapta pe proiectul <u>http://localhost/BookWeb</u>, alegeti add reference, click pe tab-ul Browse si mergeti in folder-ul C:\Book\BookDll.

Pe frmPublisherList.aspx apasati F7 si puneti urmatoarele linii in evenimentul Page\_load

IsPostBack) BindData();

Functia BindData arata la fel ca cea de pe Windows Forms:

private void BindData()

{

```
BookObjects.ColPublisher publishers = new BookObjects.ColPublisher();
publishers.Load();
grdPublisher.DataSource = publishers;
grdPublisher.DataBind();// main difference ASP si Windows forms - this
}
```

# Acum, daca rulati proiectul, o eroare o sa se intimple : **Object** reference not set to an instance of an object

Cauza este ca nu am adaugat un fisier web.config care sa contina(asemanator app.config) conexiunea la baza de date.Adaugati un fisier nou de tipul "web configuration file", acceptati numele ales de Web.Config si puneti urmatoarele linii in connection strings:

```
<connectionStrings>
```

```
<add name="SQLServer"
connectionString="server=.;Trusted Connection=true;database=Book;"/>
```

</connectionStrings>

Va trebuie sa specificam ca ne conectam la SQL Server.

<appSettings>

</appSettings>

<add key="DatabaseUsed" value="SQLServer"/>

-- possible values : MDB, SQLServer-->

Apasati din nou CTRL+F5. Eroarea aparuta acum este :

# Cannot open database "Book" requested bythelogin.Theloginfailed.Login failed for user '<PCNAME>\ASPNET'.

De ce aceasta eroare ? In connection strings am pus **Trusted\_Connection=true** in web.config. Asa incit utilizatorul care se conecteaza este cel sub care ruleaza site-ul(<PCNAME>/ASPNET).

Avem mai multe solutii :



- Sa punem in web.config o conexiune care sa foloseasca autentificarea cu user name si password (de obicei buna pentru siteurile Internet)
- Sa punem site-ul sa ruleze sub alt user- unul care sa aiba drepturi de conectare la Baza de date sites) -" vezi imaginea
- 3) Sa punem utilizatorii sa se autentifice la conectare si sa dam drepturi lor sa se conecteze – utilizand "Integrated windows" impreuna cu o setare in fisierul web.config <identity impersonate="true"/>(bun pentru site-uri intranet)
- 4) Sa cream un utilizator SQL Server pentru (IUSR\_ )care sa aiba drepturi pe baza de date.

Puteti sa va faceti alegerea - personal, prefer prima solutie.

OK.Data viitoare vom vedea cum sa inseram date in baza de date Sql Server.

Tutorial programare in .NET	г		Ignat Andrei	
BookWeb	C:\Book\BookWeb	BookWeb Properties		
, DOOKWED	Authentication Methods	HTTP Headers	Custom Errors	ASP Directory S
	Anonymous access No user name/password requ Account used for anonymous User name:	ired to access this resource. access: Browse	tion control ccess and edit the ds for this resource.	Edit
	Password: Allow IIS to	control password	to this resource using et domain names.	
	Authenticated access For the following authenticatio are required when - anonymous access - access is restricted	on methods, user name and password is disabled, or using NTFS access control lists		Edi
	Digest authentication for \	Vindows domain servers	unications and es when this	Server Certificati
	Basic authentication (pass	sword is sent in clear text)		View Certificate
	Default domain:	Select		Edi
	Realm:	Select		
	Integrated Windows authors	entication		<u> </u>
	OK	Cancel Help	Lancel	Apply
Lecturi recomandate	ervicille			
Master Pages				
<u>ConnectionStrings</u> <u>Programarea in .N</u>	<u>ET - partea a 8-a - e</u>	editarea in ASP.NET		
Acum vom edita in cadru	l proiectului Web obiecto	ele Publisher.		
Adaugati un nou WebFor separate file" si "Select m	m , numiti-o frmPublish aster page" sunt amindo	er_Insert.aspx si asigurati- va c ua selectate.	a "Place code in	

Scimbati in "source view" titlul de la "Untitled Page" la "Insert Publisher"

Acum trebuie sa punem controalele pentru inserare, adica numele si site-ul Publisher-ului.

Prefer sa pun o tabela(desi altii prefera CSS) si codul paginii o sa arate asa:

```
<% Page Language="C#" MasterPageFile="~/Book.master" AutoEventWireup
                                                            rue"
CodeFile="frmPublisher Insert.aspx.cs" Inherits="frmPublisher Insert"
Title="Insert Publisher" %>
<asp:Content ID="Content1" ContentPlaceHolderID="ContentPlaceHolde
Runat="Server">
   Enter values
          Name
          asp:TextBox ID="txtName" runat="server">
               asp:TextBox>
       Site
```

Ignat Andrei

<asp:TextBox ID="txtSite" runat="server">

</asp:TextBox>

<asp:Button ID="btnSave" Text="Insert" runat="s

<asp:Button ID="btnCancel" Text="Cancel" runat="server" />

</asp:Content>

Acum faceti click pe "design view" si faceti click dublu pe butonul "Insert" ca sa generati evenimentul de Click. Iarasi Click dublu in "solution explorer" pe frmPublisher\_Insert.aspx si, in Design view, click dublu pe butonul Cancel.

Pentru "Cancel" e clar ce trebuie facut- redirectionata pagina la frmPublisherList.aspx

Response.Redirect("frmPublisherList.aspx", false);

Pentru buton-ul de "save" va trebui sa creeam un nou "publisher" si sa il salvam :

```
Publisher p = new Publisher();
p.Name = txtName.Text;
p.Site = txtSite.Text;
p.Insert();
Response.Redirect("frmPublisherList.aspx", false);
```

Sa incercam codul setind frmPublisher\_Insert.aspx ca ,,start page" si sa rulam proiectul (F5)

Daca totul a mers bine (fiti sigur ca Insert cheama Save()) o sa vedeti in frmPublisherList.aspx exact numele si site-ul publisher-ului creat.

Este clar ca frmPublisherList.aspx are nevoie de un buton cu ajutorul caruia sa navigam pe pagina de creare.Sa il punem:

<asp:Button ID="btnNew" runat="server" Text="New" OnClick="btnNew Click" />

```
Si sa scriem codul :
```

```
protected void btnNew_Click(object sender, EventArgs e)
```

```
{
    Response.Redirect("frmPublisher_Insert.aspx", false
}
```

Atita este de ajund pentru crearea de un nou publisher.

Pentru editare si stergere se poate folosi chiar grid-ul - dar prefer sa avem doua pagini noi.

Asa ca o sa adaugam pe grid un link de edit si un buton de delete – asta pentru a vedea codul diferit generat de cele doua.

Pagina arata asa:

```
<%@ Page Language="C#" MasterPageFile="~/Book.master" AutoEventWireup="true"
CodeFile="frmPublisherList.aspx.cs" Inherits="frmPublisherList"
Title="Publisher Lists" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">
```

```
<asp:GridView ID="grdPublisher" runat="server"
AutoGenerateColumns="false">
```

<Columns>

```
asp:BoundField DataField="Site" HeaderText="Site" />
```

<asp:BoundField DataField="Name" HeaderText="Name" />

<asp:TemplateField HeaderText="Operations">

<ItemTemplate>

```
<asp:Button runat="server" ID="btnDelete"
CommandName="deletepub" CommandArgument='<%# Eval("IDPublisher") %>'
Text="Delete" />
```

#### Ignat Andrei

```
<asp:HyperLink runat="server" ID="hkEdit" NavigateUrl='<%#
Eval("IDPublisher","~/frmPublisher_Edit.aspx?ID={0}") %>'
Text="Edit"></asp:HyperLink>
```

</ItemTemplate>

</asp:TemplateField>

</Columns>

</asp:GridView>

<br />

```
<asp:Button ID="btnNew" runat="server" Text="New" OnClick="btnNew_Click" />
```

</asp:Content>

Link hkEdit este clar – redirecteaza pagina la frmPublisher\_Edit.aspx cu ID-ul publisherului de pe rindul respectiv.

Pentru buton trebuie sa cream evenimentul - si evenimentul este chiar pe grid si se numeste RowCommand

In fisierul .cs:

{

```
protected void grdPublisher_RowCommand(object sender,
GridViewCommandEventArgs e)
```

switch(e.CommandName)

se

{

"deletepub":

int idPublisher;

if(int.TryParse(e.CommandArgument.ToString(),out idPublisher))

Response.Redirect("frmPublisher\_Delete.aspx?ID="+
).

idPublisher, false);

}

}

## Ignat Andrei

return;

}

Response.Write("Can not find id:" + idPublisher);

break;

default:

Response.Write("Do not know command : " + e.CommandName);

break;

Acum sa creeam cele doua pagini noi frmPublisher Delete si frmPublisher Edit .

La amindoua o sa copiem tabela de la pagina de "new: si sursa –fara definitia de clasa. Un singur lucru e de facut : sa regasim obiectul "Publisher" dupa id-ul trimis.Mai intii regasim ID-ul :

```
int idPublisher;
if(!int.TryParse(Request,QueryString["ID"],out idPublisher))
{
    Response.Redirect("frmPublisherList.aspx", false);
    return;
}
//we have id of the publisher
```

Cum facem acum regasirea dupa ID ? Amintiti-va ca in aplicatia Windows forms ceea ce am trecut de la o forma la alta a fost chiar obiectul. Aici avem doar Id-ul. Va trebui sa adaugam acest cod de regasire al obiectului- vom deschide solutia Book.sln si vom adauga metoda respectiva.

Imi place sa pun metoda pe clasa ColPublisher si sa o fac statica – pentru ca nu depinde de vre-o variabila a clasei ColPublisher.

```
Tutorial programare in .NET
                                                                                 Ignat Andrei
public static Publisher sLoadFromID(int ID)
         {
              DbConnection db = Settings.TheConnection;
              using (db)
              {
                   db.Open();
IDataReader ir = Settings.Load("select IDPublisher,
NamePublisher, SitePublisher from Publisher where IDPublisher="+ ID, db);
                  while (ir.Read())
                   {
                       Publisher p = new
                                            Publishe
                       p.FillObject(ir);
                       return p;
                   }
              }
              return null
         }
Compilati si inapoi la proiectul Web. Acum putem folosi metoda:
//we have id of the publisher
          Publisher p = ColPublisher.sLoadFromID(idPublisher);
         if (p == null) //maybe someone deleted
         {
              Response.Redirect("frmPublisherList.aspx", false);
              return;
```

## Ignat Andrei

```
}
if (!IsPostBack)
{
    //now fill the text boxes
    txtName.Text = p.Name;
    txtSite.Text = p.Site;
}
```

De ce am pus (!IsPostBack)? Din cauza ca "textboxes" trebuie sa fie umplute cu numele Publisher-ului doar prima oara. Cind user-ul modifica numele si/sau site-ul si dupa ce da click pe "save" trebuie sa ii pastram modificarile.

Pentru a regasi usor obiectul, sa il punem intr-o proprietate:

```
private Publisher pub
{
    get
    {
        int idPublisher;
        if (!int.TryParse(Request.QueryString["ID"], out idPublisher))
        {
            return null;
            //we have id of the publisher
            return ColPublisher.sLoadFromID(idPublisher); ;
        }
}
```

}

Codul pe PageLoad va fi acum mai mic:

```
protected void Page Load(object sender, EventArgs e)
    {
        Publisher p = pub;
        if (p == null)
        {
            Response.Redirect("frmPublisherList.aspx",
            return;
        }
        if (!IsPostBack)
        {
             //now fill the text box
             txtName.Text = p.Name;
             txtSite.Text
                             p.Site;
    }
Si va trebui sa modificam si codul de pe "save" :
protected void btnSave_Click(object sender, EventArgs e)
    {
        Publisher p = pub;
        if (p == null)
```

{

}

## Ignat Andrei

```
//TODO : throw an exception that someone deleted the publisher
Response.Redirect("frmPublisherList.aspx", false);
```

return;

p.Site = txtSite.Text;

p.Name = txtName.Text;

p.Update();

Response.Redirect("frmPublisherList.aspx", false);

```
Va trebui sa modificati si text-ul butonului de la "Insert" la "Save"
```

Pe pagina de "delete" o sa punem acelasi cod ca sa regasim "Publisher"-ul. Iata codul:

```
using System;
```

}

```
using System.Data;
```

using System.Configuration;

using System.Collections;

```
using System.Web;
```

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

```
Tutorial programare in .NET
                                                                         Ignat Andrei
using System.Web.UI.HtmlControls;
using BookObjects;
public partial class frmPublisher Delete : System.Web.UI.Page
{
    private Publisher pub
    {
        get
        {
            int idPublisher;
            if (!int.TryParse(Request.QueryString[
                                                            out idPublisher))
            {
                 return null;
            }
            //we have id of the publisher
                    ColPublisher.sLoadFromID(idPublisher); ;
            return
           ed void Page_Load(object sender, EventArgs e)
        Publisher p = pub;
        if (p == null)
        {
            Response.Redirect("frmPublisherList.aspx", false);
```

```
Ignat Andrei
```

```
return;
    }
    if (!IsPostBack)
    {
        //now fill the text boxes
        txtName.Text = p.Name;
        txtSite.Text = p.Site;
    }
}
protected void btnSave_Click(object sender, Ev
                                                        e)
{
    Publisher p = pub;
    if(p != null)
        p.Delete();
    Response.Redirect("frmPublisherList.aspx", false);
}
             id btnCancel_Click(object sender, EventArgs e)
protecte
   Response.Redirect("frmPublisherList.aspx", false);
}
```

}

Nu uitati sa schimbati textul de la btnSave in "Delete". Puteti de asemenea sa puneti " ReadOnly="true" pe textboxes – ca sa nu dati impresia ca se editeaza ceva.

```
<%@ Page Language="C#" MasterPageFile="~/Book.master" AutoEventWireup="true"</pre>
CodeFile="frmPublisher_Delete.aspx.cs" Inherits="frmPublisher Delete"
Title="Untitled Page" %>
<asp:Content ID="Content1" ContentPlaceHolderID="ContentPlaceHolder1"
Runat="Server">
   Enter values
          Name
          xtName" runat="server" ReadOnly="true">
              <asp:TextBo
              </asp
             Site
          <asp:TextBox ID="txtSite" runat="server" ReadOnly="true">
              </asp:TextBox>
```

#### Ignat Andrei

<asp:Button ID="btnSave" Text="Delete" runat="server" OnClick="btnSave Click" />

<asp:Button ID="btnCancel" Text="Cancel" runat="server" OnClick="btnCancel\_Click" />

</asp:Content>

## Programarea in .NET - partea a 9-a -site map si localization

Sitemap-ul se foloseste pentru ca utilizatorul sa vada oricind "unde" este si unde poate reveni. Folosirea lui e relativ usoara:

Adaugati un nou "item" – gasiti "Site Map" si acceptati numele de Web.sitemap

Puneti urmatorul cod inauntru :

<?xml version="1.0" encoding="utf-8" ?>

<siteMap xmlns="http://schemas.microsoft.com/AspNet/SiteMap-File-1.0" >

<siteMapNode url="default.aspx" title="Main" description="First Page">

```
siteMapNode url="frmPublisherList.aspx" title="All publishers"
description="Publishers list" >
```

<siteMapNode url="frmPublisher\_Insert.aspx" title="New Publisher"
description="Add new"></siteMapNode>

```
<siteMapNode url="frmPublisher_Edit.aspx" title="Edit Publisher"
description="Edit"></siteMapNode>
```

<siteMapNode url="frmPublisher\_Delete.aspx" title="Delete Publisher"
description="Delete"></siteMapNode>

</siteMapNode>

```
<siteMapNode url="frmBookList.aspx" title="All books" description="Book
list" >
```

</siteMapNode>

</siteMapNode>

</siteMap>

(Numele sunt destul de descriptive – url, title si description)

Acum e timpul sa il vedem la lucru:

Deschideti Book.master( care tine cum arata site-ul in general), si puneti un control "site map" ( il gasiti in tab-ul "navigation" de pe "toolbox") inainte de "content place holder":

```
<asp:SiteMapPath ID="SiteMapPath1" runat="server" Font-Names="Verdana" Font-
Size="0.8em" PathSeparator=" : ">
```

<PathSeparatorStyle Font-Bold="True"

ForeColor="#990000" />

<CurrentNodeStyle ForeColor="#333333" />

NodeStyle Font-Bold="True" ForeColor="#990000" />

RootNodeStyle Font-Bold="True"

ForeColor="#FF8000"

</asp:SiteMapPath>

Si sa punem si un "tree view" in loc de "right menu":

```
<asp:TreeView ID="TreeView1" runat="server" DataSourceID="SiteMapDataSource1"
MaxDataBindDepth="1">
```

</asp:TreeView>

<asp:SiteMapDataSource ID="SiteMapDataSource1"</pre>

runat="server" />

Acum, daca veti rula proiectul si va duceti pe "new" veti vedea urmatoarele:

<b>⊟ Main</b> All publishers All books	Main : All publishers : New Publ Enter values Name Site Insert Cancel	isher
		0.

Sigur ca linkul "All books" nu e implementat – dar ramine in sarcina dvoastra sa il faceti.

Acum sa trecem la localizare. Vrem sa fim capabili ca cei care vin sa poata alege intre Engleza si Franceza

O sa facem acest lucru pentru o singura forma, iar celelalte o sa le lasam ca exercitiu.

O sa salvam setarea limbajului intr-un "cookie" pe PC-ul user-ului si o sa fie citit de fiecare data cind user-ul intra pe site.

Sa adaugam un "drop down list" pe "Book.master" :

```
<asp:DropDownList runat="server" id="ddlLanguage"
OnSelectedIndexChanged="ddlLanguage_SelectedIndexChanged" AutoPostBack="true">
```

<asp:ListItem Text="English" Value="en">

</asp:ListItem>

<asp:ListItem Text="French" Value="fr">

#### </asp:ListItem>

</asp:DropDownList>

In pagina.cs sa salvam setarea :

{

}

protected void ddlLanguage\_SelectedIndexChanged(object sender, EventArd

```
HttpCookie cookie = Request.Cookies["Language"];
```

cookie.Value = ddlLanguage.SelectedValue;

Response.AppendCookie(cookie);

cookie.Expires = System.DateTime.Now.AddYears(1);

Response.Redirect (Request.Url.LocalPath)

```
Acum, ca am salvat-o, sa o regasim:
```

```
protected void Page_Load(object sender, EventArgs e)
```

if (!IsPostBack)
ChangeLanguage();

```
}
```

private void ChangeLanguage()

```
{
```

HttpCookie cookie = Request.Cookies["Language"];

## Ignat Andrei

```
if (cookie == null)
{
    //set default the cookie in web.config
    string s = Thread.CurrentThread.CurrentUICulture.Name;
   cookie = new HttpCookie("Language");
   cookie.Value = s;
   cookie.Expires = System.DateTime.Now.AddYears(1);
   Response.AppendCookie(cookie);
}
foreach (ListItem li in ddlLanguage.Items
{
    if (li.Value == cookie.Value
    {
          li.Select
```

```
Ignat Andrei
```

Ca sa schimbam ar trebui sa scriem cod in fiecare pagina pe metoda InitializeCulture, sau sa o punem in global.asax file( care prinde evenimentele aplicatiei ) pe Application\_BeginRequest: (adaugati "new item" => "Global Application Class")

```
protected void Application_BeginRequest(object sender, EventArgs e)
```

```
{
```

}

```
string lang =
System.Threading.Thread.CurrentThread.CurrentUICulture.Name;
```

HttpCookie cookie = Request.Cookies["Language"];

if (cookie != null && cookie.Value != null)

lang = cookie.Value;

```
System.Threading.Thread.CurrentThread.CurrentUICulture =
System.Globalization.CultureInfo.GetCultureInfo(lang);
```

```
System.Threading.Thread.CurrentThread.CurrentCulture =
System.Globalization.CultureInfo.CreateSpecificCulture(lang);
```

In sfirsit trebuie sa creeam textele in franceza si engleza.

Adaugati un folder Asp.NET, numit "App\_LocalResources"



Tutorial programare in .NE	т		<u>Ignat Andrei</u>
✓ App_LocalResout.aspx.tr.resx [ ᡂ Strings ▼ ]] Add <u>R</u> esource ▼ ×	App_LocalResout.aspx.en.resx	App_LocalResoList.aspx.resx	<b>₹</b> X
Name 🛆	Value	Comment	
btnNew.Text	Nouveau		
*			
Iar adul da na hutanul 1	Naw" sa va imbogati au		269
iai codui de pe butoliui "i	New se va intogati cu	meta:resourcekey="btnnew".	
<asp:button bt<="" id="btnN&lt;br&gt;meta:resourcekey=" td=""><td>ew" runat="server" ] nNew"/&gt;</td><td><pre>"ext="New" OnClick="btnNew_C</pre></td><td>lick"</td></asp:button>	ew" runat="server" ] nNew"/>	<pre>"ext="New" OnClick="btnNew_C</pre>	lick"
Si, daca rulati aplicatia s butonului schimbindu-se	i schimbati in dropdown	din Engleza in Franceza o sa vede	ti textul
Observatie 1: Daca nu av	veti fisierul .resx fara a sj	pecifica limbanu merge!	
Observatie 2: Daca aveti putei adauga resurse resx App_GlobalResources fis	mai multe text IDENTIC in folder-ul App_Globa sierele :	E de translatat (exemplu : butonu IResources. Exemplu : presupuner	ıl de "save") n ca avem in
Buttons.en.resx	N		
Buttons.fr.resx	S.		
Buttons.resx			
Si vrem sa modificam btr	nSaveText		
Accesam resursele astfel	:		
<mark>&lt;%</mark> \$ Resources:Button	s,btnSaveText <mark>%&gt;</mark>		

## Ignat Andrei

<asp:Button ID="btnSave" Text="<mark><%</mark>\$ Resources:Buttons,btnSaveText<mark>%></mark>" runat="server" OnClick="btnSave Click" />

Sau, daca vreti in mod programatic:

Resources.Buttons.btnSaveText

Lecturi recomandate:

Localization: http://quickstarts.asp.net/QuickStartv20/aspnet/doc/localization/localization.aspx

Master Pages:

http://quickstarts.asp.net/QuickStartv20/aspnet/doc/masterpages/default.aspx

## **Programarea in .NET - partea a 10-a: Un proiect Dos**

Deschideti solutia book.sln si adaugati un nou proiect de tipul ConsoleApplication (nume : BookDos)

Adaugati o referinta la proiectul BookObjects si adaugati un fisier app.config si scrieti in el aceleasi lucruri ca si in app.config file din proiectul BookWin.

La fel pentru evenimentul de "post build" - deoarece trebuie sa copiem fisierul mdb :

copy \$(ProjectDir)..\BookData\\*.mdb \$(TargetDir)

In Program.cs file scrieti urmatorul cod:

using System;

using System.Collections.Generic;

using System.Text;

using BookObjects;

namespace BookDos

{

class Program

}

```
{
static void Main(string[] args)
{
    ColPublisher col = new ColPublisher();
    col.Load();
    Console.WriteLine("Records Number:" + col.Count);
    foreach (Publisher p in col)
    {
        Console.WriteLine(p.Name);
    }
}
```

Si, desigur, numar de inregistrari va fi 0- caci nu avem nici o inregistrare.

La fel ca in proiectul Windows, puteti creea un nou Publisher, sterge, etc.

## Programarea in .NET - partea a 11-a Teste automate cu NUnit

De ce ar trebui sa faceti teste automate?

Din mai multe motive :

- 1. Pentru ca e o modalitate usoara de a releva functionalitatile mari ale aplicatiei
- 2. Pentru ca la orice modificare la care nu sunteti sigur daca dauneaza cumva logicii aplicatiei puteti rula testele vechi si vedeti daca ati stricat ceva sau nu(Nota : ar trebui sa adaugati un nou test pentru cei care vin dupa voi )

3. E mai usor de fixat bug-urile daca, pe deasupra, rulati testele in fiecare noapte – si a doua zi dimineata vedeti ceva stricat...

4.

Hai sa trecem la treaba:

Mai intii downloadati NUnit de la <u>http://www.nunit.org/index.php?p=download</u> ( eu am folosit versiunea 2.2.8 )Exista si surse si setup de instalare. Eu as sfatui sa luati sursele sa le compilati.

Apoi la solutia noastra Book.sln adaugam un nou proiect de tipul Class Library, numit BookTest, adaugam o referinta la nunit.framework.dll, aflat in NUnit-2.2.8src\src\NUnitFramework\framework\bin\Debug2005, modificam class1.cs in TestPublisher.cs si incepem sa scriem testul.Testul cel mai simplu este unul de CRUD – create, read, update, delete.

Avem nevoie de obiectele Publisher respective, precum si de setari in fisierul App.Config pentru a recunoaste Baza de date, precum si de Baza de date.

Pentru Publisher, adaugam o referinta la BookObject in tab-ul "Projects" de la Add Reference.

Pentru App.Config, adaugam un fisier de tipul application configuration file si copiem de la BookDos partile relevante, astfel incit fisierul arata astfel :

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
<appSettings>
<add key="DatabaseUsed" value="MDB"/>
<!-- possible values : MDB, SQLServer-->
</appSettings>
<connectionStrings>
<add name="MDB" connectionString="Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=[DataDirectory]\book.mdb;User Id=admin;Password=;"/>
<!-- TODO : add for asp.net application the connection string with SQL Server-->
```

</connectionStrings>

Ignat Andrei

## </configuration>

Pentru baza de date, e simplu : in Build Events, la Post Build Command Line adaugam

```
copy $(ProjectDir)..\BookData\*.mdb $(TargetDir)
```

Acum putem incepe sa scriem testul :

using System;

using System.Collections.Generic;

using System.Text;

using NUnit.Framework;

using BookObjects;

namespace BookTest

{

[TestFixture] //arata ca e o clasa care contine teste

public class TestPublishe

{

[Test] //arata ca metoda care urmeaza este un test

[Category("CRUD")] //categoria -de obicei, testele de CRUD ar trebui puse impreuna

public void CRUD()

Publisher p = new Publisher();

p.Name = "Amazon";

p.Save();

```
Ignat Andrei
```

```
//sa il gasim
ColPublisher cp = new ColPublisher();
cp.Load();
bool bFound= false;
foreach (Publisher pLoop in cp)
{
if (pLoop.Name == p.Name)
{
bFound = true;
break;
}
}
//daca acea conditie(bFound) nu e true, atunci se afiseaza mesajul de eroare
Assert.IsTrue(bFound, "Nu s-a gasit publisher cu numele " + p.Name + " dupa
insert");
//sa il modificam
p.Name = "O'Reilly";
p.Update();
//sa il gasim din
                  nou
               blisher();
cp = new
            P
cp.Load();
bFound = false;
foreach (Publisher pLoop in cp)
{
if (pLoop.Name == p.Name)
```

```
{
bFound = true;
break;
}
}
//daca acea conditie(bFound) nu e true, atunci se afiseaza mesajul de eroare
Assert.IsTrue(bFound, "Nu s-a gasit publisher cu numele " + p.Name
update");
//acum sa il stergem
p.Delete();
//si sa vedem ca nu a fost gasit
cp = new ColPublisher();
cp.Load();
bFound = false;
foreach (Publisher pLoop
{
if
   (pLoop.Name
                   p.Name
{
bFound
break;
}
}
//daca acea conditie(bFound) nu e false, atunci se afiseaza mesajul de eroare
```

}

}

}

## Ignat Andrei

```
Assert.IsFalse(bFound, "S-a gasit publisher cu numele " + p.Name + " dupa
delete");
```

Il compilam si sa rulam testul. Gasiti in folderul NUnit-2.2.8-src\src\GuiRunner\nunit-guiexe\bin\Debug2005 un nunit-gui.exe si porniti-l.Apasati File=> Open si mergeti in C:\Book\BookTest\bin\Debug si incarcati BookTest.dll. Ar trebui sa apara figura urmatoare

- Party and a second second		
BookTest.dll - NUnit		
Elle View Project Iools Help		
Tests Categories		
C\Book\BookTest\bin\Debug\BookTest.dll BookTest TestPublisher CRUD	Bun Stop CRUD	
	Errors and Failures Tests Not Run Console Out Console Error	
		*
		×
		-
		Ľ
L		4
CRUD	Test Cases : 1	1.

Apasati pe RUN si primul lucru pe care il vedeti este:

Calegores     Calegores     Cosol/Book/Est/Debug/Book/Est/dl     Book/Est     Sol/Test*Usher	Bun	Gree CRUD				
Lender.	Errors and Failures	lests Not Run Console.Out Co	nole Enn	to the second second		
	41					
	at BookTes	ects.Publisher.Opdat t.TestPublisher.CRUD	s() in C:\Book\B () in C:\Book\Bo	ookObjecta\Pubj okTest\TestPubj	isher.cs:line isher.cs:line	107 37
	1					

Se vede clar ca e ceva prost... ne uitam in TestPublisher.cs si vedem ca la linia 37 este

p.Update();

Ceva a mers prost la update ... sa vedem linia 107 din Publisher.cs

strSQL += "'" + this.Site.Replace("'", "''") + "'";

Acum e clar ce s-a intimplat... Cind am facut testul, nu am initializat Site-ul cu nimic... si atunci este null, ceea ce inseamna ca .Replace nu poate fi aplicat

Sa modificam codul din Publisher.cs ca sa ia in seama si acest lucru :

```
if (this.Site == null)
strSQL += " NULL ";
else
strSQL += "'" + this.Site.Replace("'", "''") + "'";
```

Acum apare alta eroare :
## kTestidil - NUni Categories C:\Book\BookTest\bin\Debug\BookTest.dll CABook \Book Test \bin\Debug\Book Test d Bur ok.Les TestPublishe CRUD Errors and Failures | Tests Not Run | Console Out | Console Error 88 at BookTest.TestPublisher.CRUD() in C:\Book\BookTest\TestPublisher.cs:line 52 Test Cases 1 Time: 0.8906136 Completed Tests Run : 1 Falues: 1

Este destul de clar ca aplicatia nu a updatat numele ... «

Tutorial programare in .NET

De ce ?Ne dam seama imediat : in momentul in care aplicatia a adaugat un nou Publisher, nu a regasit ID-ul inserat ... iar cind a facut update, IDPublisher este 0, ceea ce inseamna ca nu a putut fi facut update corect.

Cum modificam acest lucru? Pentru access, putem sa selectam maxim de ID, iar pentru SQL Server putem crea o procedura stocata ... sau sa intoarcem @@Identity

Hai sa facem pentru Access, modificand Publisher.cs, procedura Save, adaugind la final:

if(Settings.TheDatabase == Settings.DatabaseUsed.MDB)

{

strSQL = "select max(IDPublisher) as nr from Publisher"; using (DbConnection dc = Settings.TheConnection)

```
dc.Open();
```

using(DbCommand dco =Settings.TheCommand)

{

{

## Ignat Andrei

<pre>dco.CommandType = System.Data.CommandType.Text;</pre>	
dco.CommandText = strSQL;	
dco.Connection = dc;	
<pre>object o = dco.ExecuteScalar();</pre>	
<pre>this.IDPublisher = int.Parse(o.ToString());</pre>	
}	
}	
}	

Acum rulam din nou testul si totul e verde, ceea ce e de bine :

CNBook/BookTest/bin/Debug/BookTest.dl     SockTest     SockTest     TestPublisher     CRUD	Bun	Grop	C.\Book\BookTest	ber/Debug/Blook Test				
	Errors and Failures Tests Not Run Console Out Console Error							
	1							
	4							
	a							
ried				Test Cases : 1	Tests Run : 1	Falues: 0	Tme: 0.937494	
					ACC -	1000000	1	

E clar ca exemplu a fost mai degraba simplut, iar ceea ce conteaza, de fapt, sunt regulile de business si de validare - ca de exemplu, validarea CNP

# Programarea in .NET - partea a 12-a Documentarea - Scrierea de fișiere Help.

In .NET sunt doua tipuri mari de fișiere Help : Cele care produc Help pentru utilizatorul final si cele care sunt auto-generate din comentarii la cod.

Avem nevoie de următoarele:

1.Html Help Workshop – e free si puteți sa îl downloadati de <u>aici</u>

http://www.microsoft.com/downloads/details.aspx?familyid=00535334-c8a6-452f-9aa0d597d16580cc&displaylang=en. Adițional puteți downloada si fișiere css stil Office de <u>aici</u>

http://www.microsoft.com/downloads/details.aspx?FamilyId=A6A76073-0E0A-49BB-8E21-318B798B4CF6&displaylang=en

2. Pentru documentația codului exista înainte NDoc – dar din păcate dezvoltatorul nu mai face dezvoltarea pentru .Net 2.0( vezi <u>http://johnsbraindump.blogspot.com/2006/07/ndoc-20-is-dead.html</u>)

Alternativa este SandCastle (<u>http://www.sandcastledocs.com</u>) din care ultimul CTP(Martie) este <u>aici (http://www.microsoft.com/downloads/details.aspx?FamilyID=E82EA71D-DA89-42EE-A715-696E3A4873B2&displaylang=en</u>).

3.De asemenea, pentru ca SandCastle e greu de utilizat din command line, exista mai multe GUIuri pentru el – intre care vom lucra cu SandCastle Help File Builder de pe CodePlex (http://www.codeplex.com/Wiki/View.aspx?ProjectName=SHFB). O sa lucram cu ultimul release ,1.4.0.1 PROD aflat aici

http://www.codeplex.com/SHFB/Release/ProjectReleases.aspx?ReleaseId=2264

si as downloada chiar sursele ...

Începem cu documentarea codului pentru BookObjects, urmând sa trecem la generarea de Help pentru proiectul Windows.

Comentariile in C# se fac scriind trei slash-uri deasupra clasei/metodei/cimpului pe care vreți sa le documentați:

```
/// <summary>
/// Aceasta clasa tine toti publisher-ii
/// Mod de utilizare : folositi Load
/// </summary>
public class ColPublisher :
System.Collections.ObjectModel.KeyedCollection<string,Publisher
{
/// <summary>
/// varianta interna de generat cheie unica pentru
                                                          Publisher
/// </summary>
/// <param name="item">publisher-ul<
/// <returns></returns>
protected override string GetKeyForItem(Publisher item)
{
Continuați cu toate metodele sau downloadati ultima varianta de proiect de aici:
http://serviciipeweb.ro/iafblog/content/binary/part12/book.zip
```

In plus, trebuie sa mai setați faptul ca trebuie generata documentația XML din proprietatile proiectului:

1.G	•		- @	
¥,G				
			• ×	s • 9
Warning level:	14		*	Solu
Suppress warnings:				B
t warnings as errors	10			9
None				8
Specific warnings:	Ľ.			-
				8 3
Ordered nath:	bin\Release\		Brown	€ €
XML documentation file:	I			-
Register for COM interop			-	a 🖪
Generate serialization assembly:	Auto	~		€ 1 € 1
			Advan	(H- (32)
			M	
	Warning level: Suppress warnings: t warnings as errors None Specific warnings: All put Output path: WML documentation file: Register for COM interop Generate serialization assembly:	Warning level: 4 Suppress warnings: t warnings as errors None Specific warnings: All put Output path: biniRelease( XML documentation file: Register for COM interop Generate serialization assembly: Auto	Warning level: 4 Suppress warnings: t warnings as errors None Specific warnings: All put Output path: binlRelease\ VML documentation file: Register for COM interop Generate serialization assembly: Auto	Warning level: 4 Suppress warnings: t warnings as errors None Specific warnings: All put Output path: binlRelease\ Browe WML documentation file: Register for COM interop Generate serialization assembly: Auto Advan

Acum in folder-ul C:\Book\BookObjects\bin\Debug aveți generata documentația XML. Pornim SandCastle Help File Builder care arata cam asa:

torial programare in .NET			<u>Ignat An</u>
Unnamed - Sandcastle Help File Bu	rilder		
Project Documentation Help			
) 🗳 🖬 🖻 🛎 🔒 🔎 🕅 🍘			
Assemblies to Document			
		<u>Add</u>	a Eeniove
		Pri <u>S</u> ummary	Namespaces
Project Properties			
2021 00			
Additional Content			^
AdditionalContent	(Collection)		
ContentPlacement	AboveNamespaces		
🖯 Build			
CleanIntermediates	True		
ComponentConfigurations	(Collection)		
CppCommentshoup	False		
Dependencies Examinate Maxim	(Lotection) 2.0.50727		×
AdditionalContent Additional content that should appear in the h	elp file		
Dutput			
[Sandcastle Welp Tile Duilder GUI, ve	usion 1.3.4.0]		
<b>start</b> 🕅 Unnamed - Sandcasti	an l		🔮 🔍 🧐 6.54 AM

Apăsam pe Add si ne ducem in C:\Book\BookObjects\bin\Debug. Acolo indicam dll-ul generat iar programul o sa "observe" si fișierul XML. Apăsam pe iconița de compilare si ... eroare...

Error: Unresolved assembly reference: System.Configuration (System.Configuration, Version=2.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a) required by BookObjects

Last step completed in 00:00:02.403

După descriere vedem că ii lipsește o referința la System.Configuration. Acest dll se afla in GAC (Global Assembly Cache) si o vom adăuga. În Project Properties, la Build=>Dependencies apăsați pe butonul cu cele 3 puncte.

Acum in ecranul următor exista un buton cu imaginea de folder si cu o icoana de o cheie care iese, iar la tooltip scrie "Add GAC dependencies"

Tutorial programare in .NET	Ignat Andrei
Add GAC dependencies	Close
Start Book - Sandcastle Hel	248 2 9 9 7:03 AM
Apăsați pe el si căutați System.Configuration, Version=2.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a in lista care urmează.	

Acum rulati din nou si ceea ce se va genera este un document chm, numit Documentation.chm aflat in C:\Book\BookObjects\bin\Debug\Help:

ide Locate Back Forward St	op Refresh	Home	Print Options			10100
ntents   Index   Search	A Sandcast ColPu Namesp	de Docume blishe aces > l	inted Class Library F Class BookObjects > 0	ColPublisher	[C#	3
CoPublisher Constructor GetKeyForitem Method (ite Load Method	Aceasta E Synt	a clasa ti tax	ne toti publisher-	-ii Mod de utilizare : fo	olositi Load	
sLoadFromID Method (ID)	C#		Visual Basic	Managed C++		
Settings DatabaseUsed Enume	E Men	nbers	Constructors	Methods	Properties	Fie
	Public Prot	ic ected		₽ Instance ₽ Static		<u>द</u> द
	Icon	Memt	ier	Description		
	•	ColPu	blisher()	Initializes a new ins class.	tance of the ColPu	blisher
	*	Add(P	Publisher)	Adds an object to t Collection <t>. (Inherited from Col</t>	he end of the lection <publisher:< td=""><td>&gt;.)</td></publisher:<>	>.)
	100433	1922 20000		Observes the local se	concisted with the c	necified

Putem personaliza ceea ce se generează destul de ușor, modificând setările de aici:

book - Sandcastle Help File Builder	
Project Documentation Help	And a second
) 🨅 🗔 🗠 🚠 泽 🔑 🗛 🍘	
Assemblies to Document	
BookObjects.dl, BookObjects.xml	
Project Properties	
Help File	
BinawIOC	Taxe
ConvictHuef	
Copyright not	Andrei Ignat
FeedbackEMailAddress	ignatandrei@vahoo.com
HeaderText	Help Book
HelpTitle	Help Book
HtmlHelpName	HelpBook
ISAN INCOMPANY AND TAXABLE	

Tutorial programare in .NET

Ignat Andrei

E clar acum ca aceasta documentație se poate regenera la cerere.Daca vreti help in forma 2.0(hxs), downloadati Visual Studio 2005 SDK Version 4.0 de la adresa http://www.microsoft.com/downloads/details.aspx?FamilyID=51a5c65b-c020-4e08-8ac0-3eb9c06996f4&DisplayLang=en

Sa generam acum documentația pentru Windows Forms.

Ar trebui pentru fiecare forma sa avem cate un Help – așa ca o sa cream 3 fișiere HTML care o sa tina List, Add si Update.

Vom crea un nou folder, numit HelpWindows, in C:\book\Help si vom pune in el cele 3 fișiere : add.htm,list.htm,update.htm.

Pornim HTML Help Workshop, File=>New =>Project si dați next. Adaugati fișierele htm si apăsati pe Contents. Acceptați creerea unui nou fișier si apăsând pe iconița din stânga cu aspect de fișier adăugat la conținut cele 3 fisiere, dindu-le numele corespunzătoare. Acum, după compilare, s-a generat un fișier chm. Haideți sa îl integram cu aplicația Windows.

Sarcina de a copia fișierul chm lângă executabil o las cititorului, având in vedere ca am mai făcut așa ceva(Project=>Properties=>Build Events). Sa mergem la forma de list si sa adaugam din ToolBox un control HelpProvider. LA proprietati la HelpNameSpace puneti numele chm-ului. Acum pe forma, gasiti HelpKeyword on... (setati valoarea la list.htm) si HelpNavigator on...( setat la topic). Rulati, apasati F1 si iata fisierul de help!

Puteti downloada ultimele surse de aici

http://serviciipeweb.ro/iafblog/content/binary/part12/book.zip

Lecturi utile:

GAC <a href="http://www.codeproject.com/dotnet/DemystifyGAC.asp">http://www.codeproject.com/dotnet/DemystifyGAC.asp</a>

MSHelp 2.0 http://www.helpware.net/mshelp2/h20.htm

# **Programarea in .NET - partea a 13-a - Log-area operațiilor cu log4net**

In orice aplicație este bine sa ținem evidenta operațiilor făcute de utilizator( ce a modificat sau chiar ce a văzut). In acest scop putem folosi fie mecanismul de <u>trace</u> din .net, fie o soluție proprie, fie Logging si Instrumentation Application Block(http://msdn2.microsoft.com/en-us/library/ms998162.aspx), fie log4net(http://logging.apache.org/log4net/

Vom utiliza in acest exemplu log4net .El suporta log-area operațiilor in felurite moduri – in fișier, baza de date, email, telnet si <u>multe altele</u>.

Downloadati versiunea 1.2.10 de la adresa <u>http://logging.apache.org/log4net/downloads.html</u> si sa începem configurarea aplicației. Copiați conținutul folder-ului log4net-1.2.10\bin\net\2.0\debug in C:\Book\sharedDll si sa începem modificarea proiectului Windows pentru a înregistra ce a făcut utilizatorul

Deschidem Book.sln si deschidem App.Config. Acolo scriem următoarele imediat sub configuration:

```
<configSections>
```

```
<section name="log4net"
type="log4net.Config.Log4NetConfigurationSectionHandler, log4net"/>
```

</configSections>

<log4net>

<appender name="RollingLogFileAppender"
type="log4net.Appender.RollingFileAppender">

File value="Log4Net.log"/>

<AppendToFile value="true"/>

<rollingStyle value="Composite"/>

<maximumFileSize value="1MB"/>

<maxSizeRollBackups value="10"/>

<datePattern value="yyyyMMdd"/>

```
Tutorial programare in .NET
```

```
Ignat Andrei
```

```
<layout type="log4net.Layout.PatternLayout">
```

```
<param name="ConversionPattern" value="%d [%t] %-5p %c %m%n"/>
```

```
</layout>
```

</appender>

<root>

```
<level value="Debug"/>
```

```
<appender-ref ref="RollingLogFileAppender"/>
```

</root>

</log4net>

După cum se vede, folosim RollingLogFileAppender ( adică un fișier a cărui denumire va fi diferita in fiecare zi după modelul datePattern ) de tipul Composite(daca depaseste maximumFileSize atunci se creează un nou fișier în ziua respectiva).

Sa adăugam referința proiectului nostru (BookWin) dll-ul log4net.dll din sharedDll .Avem de făcut următoarele in Program.cs :

In funcția Main scriem prima linie:

log4net.Config.XmlConfigurator.Configure();

apoi adăugam următorul membru in clasa Program:

```
public static readonly log4net.ILog logger =
log4net.LogManager.GetLogger("RollingLogFileAppender");
```

Haide sa scriem in fișier de cite ori un utilizator adaugă un nou Publisher.

In frmPublisherAdd.cs, la evenimentul private void btnAdd\_Click(object sender, EventArgs e) vom adăuga codul de log-are:

if (Program.logger.IsDebugEnabled)

Program.logger.Debug("Adaugat publisher cu numele:" + p.Name);

Cam atit e de făcut. Acum rulați proiectul, adaugați un Publisher, si o sa vedeți un fișier log4Net.log in care scrie următoarele:

<data>[1] DEBUG RollingLogFileAppender Adaugat publisher cu numele:newpub

E interesant la log4Net ca puteți adăuga mai mulți appender-i, astfel ca, de pilda, sa trimită si email de cate ori o modificare e făcuta.

Observatie 1:

Pentru aplicatia Web, modificarile in Web.Config sunt aceleasi - iar in global.asax trebuie pusa linia urmatoare:

void Application\_Start(object sender, EventArgs e)

```
{
    // Code that runs on application startup
    log4net.Config.XmlConfigurator.Configure();
}
```

Observatie 2:

In loc sa punem codul in fiecare pagina de Web si Windowspe salvare, mai bine punem in fiecare cod de "salvare" al obiectelor- de pilda in public void Save()

Lectura obligatorie: documentația de <u>log4net</u>...

# Programarea in .NET - partea a 14-a Salvare XML si Executare de operatii asincrone in Windows Forms

De ce taskuri asincrone ? In ideea ca , intr-o aplicatie Windows(si chiar ASP.NET) , operatiile lungi ar trebui sa fie executate de catre alt thread, urmind ca aplicatia sa poata sa mai afiseze ceva utilizatorulu in tot acest timp ( fie si un buton pe care scrie "apasa ca sa intrerupi operatia asta lunga …"). De pilda, in aplicatia noastra, daca avem mai mult de 100 de Publisher-i si vrem sa ii vedem pe toti – ar trebui incarcati intr-un nou thread.

Ne ocupam mai intii de o aplicatie Windows Forms si pe urma de ASP.NET

Un thread nu e greu de pornit. Hai sa vedem un exemplu:

```
System.Threading.Thread t = new System.Threading.Thread(new
System.Threading.ParameterizedThreadStart(StartAction));
```

t.Start("obiect transmis");

public void StartAction(object o)

{

string s = o.ToString();

System.Threading.Thread.Sleep(5000);

//executa actiunea

trimite text

(this.InvokeRequired)

this.Invoke((MethodInvoker)delegate()

this.Text = s;

});

{

//sau

//this.Invoke(new MethodInvoker(evenimentfaraparametri());

}

#### Ignat Andrei

```
//this.Invoke(new EventHandler(btnDiscounts_Click));
//this.BeginInvoke(new EventHandler(eveniment cu parametri));
}
else
this.Text = s;
```

Totusi, exista o problema – dintr-un thread nu se pot accesa DIRECT controale din alt thread – si de aceea avem instructiunea this. Invoke .Diferenta intre this.Invoke si este this.BeginInvoke aceea ca prima instructiune asteapta rezultatul actiunii, pe cind a doua doar executa si se intoarce imediat sa execute codul ramas.

De aceea exista controlul numit BackgroundWorker – care asigura ca, din evenimentul propriu generat, sa accesezi orice obiect de pe forma. O sa facem acest lucru pentru salvarea in XML a colectiei de Publisher-i in format XML.

.NET are o forma usoara de a salva o colectie/dasa in format XML, salvindu-i proprietatile.

Vom utiliza modalitatea cea mai usoara de a face acest lucru

Marcam clasa Publisher si clasa colectie ColPublisher cu atributul de [Serializable] :

[Serializable

public class ColPublisher :
System.Collections.ObjectModel.KeyedCollection<string,Publisher>

public class Publisher

Acum o sa facem serializarea obiectului Publisher:

#region Serializer

/// <summary>

#### Ignat Andrei

```
/// instanta pentru lazy load
        /// </summary>
        private static XmlSerializer m Serializer;
        /// <summary>
        /// serializator pentru obiectul publisher
        /// </summary>
        private static XmlSerializer Serializer
        {
            get
            {
                if (m_Serializer == null)
                    m Serializer = new XmlSerializer(typeof(Publisher));
                return m Serializer;
            }
        }
            <summary>
            salveaza obiectul
                              ca XML
        [XmlIgnore
                 ring XML
            get
                StringBuilder sb = new StringBuilder();
                EncodingStringWriter sw = new EncodingStringWriter(sb,
Encoding.Default);
                XmlTextWriter xtw = new XmlTextWriter(sw);
```

```
Ignat Andrei
```

```
Serializer.Serialize(xtw, this);
return sb.ToString();
```

```
}
```

}

/// <summary>

/// recreeeaza un Publisher dintr-un string XML  $\,$ 

/// </summary>

/// <param name="XML">string care contine tot

/// <returns></returns>

public static Publisher FromXML(string XML)

StringReader sr = new StringReader(XML);

return Serializer.Deserialize(sr) as Publisher;

#endregion

{

}

Copiem apoi acelasi cod( cu citeva diferente) si pentru ColPublisher

## Citeva comentarii despre cod:

De ce am pus [XmlIgnore] peste public string XML ? Pentru a nu serializa si aceasta proprietate, dind astfel nastere la o nedorita recursivitate

Ce e cu clasa EncodingStringWriter ? Este facuta pentru a putea schimba Encoding=ul- daca aveti de exemplu caractere speciale(diacritice) romanesti/franceze/etc.

De ce metoda FromXML este statica- iar XML este pe instanta? Asa mi se pare normal – transformarea dintr-un obiect in XML sa apartina obiectului, iar din XML in obiect nu poate sa apartina

{

}

{

unui obiect( ah, daca as fi putut scrie this = Serializer.Deserialize(sr) as Publisher !) - ci
doar clasei.

Nu se poate face codul mai "generic"? Ba da- una din deosebiri ar fi ca FromXML ar trebui sa fie pe instanta...

Haideti acum in proiectul Windows sa serializam o colectie de Publisher-i.Pe forma frmPublisherList adaugam un buton btnSave, cu textul Save, dublu click si scriem urmatorul cod:

```
private void btnSave Click(object sender, EventArgs e)
```

```
BookObjects.ColPublisher col =
colPublisherBindingSource.DataSource as BookObjects.ColPublishe
```

```
string strSave =
System.Environment.GetFolderPath(System.Environment.SpecialFolder.ApplicationD
ata);
```

```
strSave = Path.Combine(strSave, "pub.xml");
File.WriteAllText(strSave, col.XML);
System.Diagnostics.Process.Start(strSave);
```

Rulati proiectul, adaugati 2 publisher-i si apasati pe save.

Este clar ca, daca sunt multi publisher-i, procesul poate deveni prea lung si blocheaza interfata.

Haideti sa folosim background worker. Il adaugam din toolbox, il redenumim bgSave, dublu click. Luam codul din btnSave Click, il adaugam la si pe urma scriem doar bgSave.RunWorkerAsync();

```
private void btnSave Click(object sender, EventArgs e)
```

```
bgSave.RunWorkerAsync();
```

}

#### Ignat Andrei

```
private void bgSave DoWork(object sender, DoWorkEventArgs e)
```

```
{
BookObjects.ColPublisher col =
```

colPublisherBindingSource.DataSource as BookObjects.ColPublisher;

```
string strSave =
```

System.Environment.GetFolderPath(System.Environment.SpecialFolder.ApplicationD
ata);

```
strSave = Path.Combine(strSave, "pub.xml");
```

```
File.WriteAllText(strSave, col.XML);
```

System.Diagnostics.Process.Start(strSave);

Linkuri recomandate:

}

BackgroundWorker in .NET 1.1

http://weblogs.asp.net/rosherove/articles/BackgroundWorker.aspx

# Programarea in .NET - partea a 15-a Operatii asincrone in ASP.NET si AJAX

Sa discutam despre operatii asincrone in ASP.NET.

Exista doua tipuri mari de operatii operatii asincrone

1. Cele care se executa doar pentru operatii lungi, care iau ceva timp de executie, si pentru care user-ul trebuie instiintat de evolutia lor.

2. Cele care trimit rapid o cerere la server si se intorc.

Pentru 1.,o solutie recomandata este sa puneti Async=true in codul de pagina si sa executati codul cu RegisterTaskAsync . Asta este interesant, pina cind ne dam seama ca modul de afisare al paginii este acelasi – adica este intirziata pina cind se termina toata de executat.

Un exemplu bun gasiti aici

http://msdn.microsoft.com/msdnmag/issues/05/10/WickedCode/#S5

si <u>aici</u>

http://msdn.microsoft.com/msdnmag/issues/07/03/WickedCode/default.aspx

Totusi, as recomanda o alta abordare – pentru ca user-ul sa "vada" desfasurarea detaliata a actiunilor efectuate, as recomanda scrierea cu Response.Write intr-un div si ascunderea apoi a div-ului .

Ceva de genul acesta:

Pe pagina aspx, printre ultimele linii, ascunderea div-ului cu mesaje:

#### Ignat Andrei

//executare al doilea task

Response.Write("Generated al doilea task<BR>");

Response.Flush();

//incheiere procedura...

Response.Write("</div>");

Response.Flush();

Acest model se poate combina cu evenimente generate in cadrul task-ruilor astfel incit, daca task-urile dureaza prea mult, user-ul sa aiba totusi un feedback despre ce ea ce se intimpla.

Sa discutam acum despre 2, executarea de operatii rapide pe server. Sa dam un exemplu simplu, si anume cautarea de publisher dupa nume. Ar fi superb daca aplicatia noastra, la apasarea primei litere a publisher-ului, ar putea sa sugereze publisher-ii care incep cu litera respectiva.

Pentru aceasta vom folosi Ajax, si vom folosi implementarea de AutoComplete de la Ajax Control Toolkit

Mai intii , trebuie sa downloadam Ajax1.0 de aici.

Apoi va trebui sa modificam Web.Config astfel incit sa suporte Ajax.

Asta inseamna ca o sa luam o mare parte din Web.Config-ul unui site Ajax si o sa il mutam la noi in site.

Sa incepem :

De la o configuration o sa luam

```
<configSections>
```

<sectionGroup name="system.web.extensions"
type="System.Web.Configuration.SystemWebExtensionsSectionGroup,
System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35">

```
<sectionGroup name="scripting"</pre>
```

```
type="System.Web.Configuration.ScriptingSectionGroup, System.Web.Extensions,
Version=1.0.61025.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35">
```

<section name="scriptResourceHandler"
type="System.Web.Configuration.ScriptingScriptResourceHandlerSection,</pre>

#### Ignat Andrei

System.Web.Extensions, Version=1.0.61025.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35" requirePermission="false" allowDefinition="MachineToApplication"/>

<sectionGroup name="webServices"</pre> type="System.Web.Configuration.ScriptingWebServicesSectionGroup, System.Web.Extensions, Version=1.0.61025.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35">

<section name="jsonSerialization"</pre> type="System.Web.Configuration.ScriptingJsonSerializationSection, System.Web.Extensions, Version=1.0.61025.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35" requirePermission="false" allowDefinition="Everywhere"/>

<section name="profileService")</pre> type="System.Web.Configuration.ScriptingProfileServiceSectio" System.Web.Extensions, Version=1.0.61025.0, Culture=neutr PublicKeyToken=31bf3856ad364e35" requirePermission="fals allowDefinition="MachineToApplication"/>

<section name="authenticationService"</pre> type="System.Web.Configuration.ScriptingAuthenticationServiceSection, System.Web.Extensions, Version=1.0.61025.0, Culture=neutral,

PublicKeyToken=31bf3856ad364e35" requirePermission="false" allowDefinition="MachineToApplication"

</sectionGrou

</sectionGroup

</sectionGroup

</configSections

De la system.web o sa luam

<pages:

<controls>

<add tagPrefix="asp" namespace="System.Web.UI" assembly="System.Web.Extensions, Version=1.0.61025.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35"/>

</controls>

</pages>

Si

#### Ignat Andrei

<compilation debug="false">

<assemblies>

<add assembly="System.Web.Extensions,
Version=1.0.61025.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35"/>

<add assembly="System.Design, Version=2.0.0.0, Culture=neutral, PublicKeyToken=B03F5F7F11D50A3A"/>

<add assembly="System.Web.Extensions.Design, Version=1.0.61025.0, Culture=neutral,

PublicKeyToken=31BF3856AD364E35"/></assemblies>

</compilation>

Apoi Handler-e si Module:

<httpHandlers>

<remove verb="\*" path="\*.asmx"/>

<add verb="\*" path="\*.asmx" validate="false"

type="System.Web.Script.Services.ScriptHandlerFactory, System.Web.Extensions, Version=1.0.61025.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35"/>

<add verb="\*" path="\* AppService.axd" validate="false"
type="System.Web.Script.Services.ScriptHandlerFactory, System.Web.Extensions,
Version=1.0.61025.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35"/>

<add verb="GET,HEAD" path="ScriptResource.axd"
type="System.Web.Handlers.ScriptResourceHandler, System.Web.Extensions,
Version=1.0.61025.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35"
validate="false"/>

</httpHandlers>

<httpModules>

<add name="ScriptModule"
type="System.Web.Handlers.ScriptModule, System.Web.Extensions,
Version=1.0.61025.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35"/>

</httpModules>

Apoi extensions si WebServer:

<system.web.extensions>

<scripting>

<webServices>

#### Ignat Andrei

<!-- Uncomment this line to customize maxJsonLength and add a custom converter --> <!--<jsonSerialization maxJsonLength="500"> <converters> <add name="ConvertMe" type="Acme.SubAcme.ConvertMeTypeConverter"/> </converters> </jsonSerialization> --> <!-- Uncomment this line to enable the authentication service. Include requireSSL="true" if appropriate. --> <!--<authenticationService enabled="true" requireSSL = "true|false"/> --> <!-- Uncomment these lines to enable the profile service. To allow profile properties to be retrieved and modified in ASP.NET AJAX applications, you need to add each property name to the readAccessProperties and writeAccessProperties attributes. --> <profileService enabled="true"</pre> readAccessProperties="propertyname1,propertyname2" writeAccessProperties="propertyname1,propertyname2" /> /webServices> <!--<scriptResourceHandler enableCompression="true" enableCaching="true" /> --> </scripting> </system.web.extensions>

http://www.serviciipeweb.ro/iafblog

Page 95

#### Ignat Andrei

<system.webServer>

<validation validateIntegratedModeConfiguration="false"/>

<modules>

<add name="ScriptModule" preCondition="integratedMode"
type="System.Web.Handlers.ScriptModule, System.Web.Extensions,
Version=1.0.61025.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35"/>

</modules>

<handlers>

<remove name="WebServiceHandlerFactory-Integrated">

<add name="ScriptHandlerFactory" verb="\*" path="\*.asmx"
preCondition="integratedMode"
type="System.Web.Script.Services.ScriptHandlerFactory, System.Web.Extensions,
Version=1.0.61025.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35"/>

<add name="ScriptHandlerFactoryAppServices" verb="\*"
path="\*\_AppService.axd" preCondition="integratedMode"
type="System.Web.Script.Services.ScriptHandlerFactory, System.Web.Extensions,
Version=1.0.61025.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35"/>

<add name="ScriptResource" preCondition="integratedMode"
verb="GET,HEAD" path="ScriptResource.axd"
type="System.Web.Handlers.ScriptResourceHandler, System.Web.Extensions,
Version=1.0.61025.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35"/>

</handlers>

</system.webServer

Acum downloadam <u>Ajax Control Toolkit</u> si vom referentia controalele existente, aflate in AjaxControlToolkit-NoSource\SampleWebSite\Bin . Vom adauga un nou tab in ToolBox, ii vom zice AjaxControls si vom adauga itemii apasind pe Choose Items:



Si apoi indicind prin browse calea la AjaxControlToolkit.dll pe care l-am downloadat. Apasati pe urma OK si vom avea controalele Ajax.

Trageti un AutoCompleteExtender si un textbox in frmPublisherList.aspx .

#### Vom completa

public class wsPublisher : System.Web.Services.WebService {

#### cu atributul

```
[System.Web.Script.Services.ScriptService]
```

si vom adauga o metoda pentru regasirea publisher-ilor care inœp cu o litera data:

#### [WebMethod]

public string[] GetCompletionPublishers(string prefixText, int count)

#### Ignat Andrei

```
{
        if (count <= 0)</pre>
            count = 10;
        List<string> items = new List<string>(count);
        BookObjects.ColPublisher publishers = new BookObjects.ColPublish
                                                                            r();
        publishers.Load();
        foreach (BookObjects.Publisher pub in publishers)
        {
            if (pub.Name.IndexOf(prefixText,
StringComparison.CurrentCultureIgnoreCase) == 0)
                items.Add(pub.Name);
            if (items.Count == count)
                break;
        }
        return items.ToArray()
    }
O vom folosi in Autocomplete:
<asp:ScriptManager runat="server">
      asp:ScriptManager>
    <asp:TextBox ID="txtPub" runat="server" autocomplete="off"></asp:TextBox>
    <cc1:AutoCompleteExtender ID="AutoPub" runat="server"
    TargetControlID="txtPub"
```

#### Ignat Andrei

ServicePath="wsPublisher.asmx"

ServiceMethod="GetCompletionPublishers"

MinimumPrefixLength="1"

CompletionInterval="1000"

EnableCaching="true"

CompletionSetCount="12">

</ccl:AutoCompleteExtender>

In momentul de fata putem testa aplicatia si vedea ca se listeaza numele publisher-ilor.

Lucruri de facut:

Creat o metoda prin care sa se poata incarca doar cei care au prefix, nu toti publisher-ii cum am facut in public string[] GetCompletionPublishers(string prefixText, int count)

De pus scriptManager-ul in Master- ca sa nu fim nevoiti sa il punem in fiecare pagina.

De vazut celelalte controale de la <u>Ajax Control Toolkit</u> (live demo la adresa <u>http://ajax.asp.net/ajaxtoolkit/Default.aspx</u>)

Data viitoare vom vorbi despre scoaterea de rapoarte.

# Programarea in .NET - partea a 16-a Rapoarte in ASP.NET

Orice aplicatie trebuie sa aiba posibilitatea de a tipari datele. Pentru aceasta in VS2005 Express se poate folosi componenta Report Viewer care e free si se poate downloada de la <a href="http://www.gotreportviewer.com">http://www.gotreportviewer.com</a>.

Dupa ce o instalati, o sa aveti in toolbox urmatoarea componenta:

Toolbox – 🗸	
E FileUpload	
🛟 Wizard	
🔣 Xml	
D MultiView	
Panel	
🔀 PlaceHolder	
C View	A)
🔁 Substitution	
🎡 Localize	
🗆 Data	
Pointer	XV
进 GridView	
🔁 DataList	
🔁 DetailsView	
- FormView	102
🔛 Repeater	
🧧 SqlDataSource	
🍺 AccessDataSource	
该 ObjectDataSource	
[ XmlDataSource	
强 SiteMapDataSource	
🔝 ReportViewer	
Validation     ReportViewer	
Navigation	
🗄 Login	
WebParts	
• HTML	
🕒 General	

Acum adaugam un raport care sa fie afisat de catre aplicatie – deschidem aplicatia Web si adaugam un nou item de tipul Report si il numim rptPublisher.rdlc :

# Ignat Andrei

	ttp://localhost/Bo	okWeb/			? ×
Templates:					000 000 000
Visual Studio	installed template	:S			
Web Form HTML Page Style Sheet Text File DataSet Mobile Web I Script File XSLT File	Form	Master Page Meb Service Web Configuration File Resource File Generic Handler Stript File Mobile Web User Control Skin File	<ul> <li>Web User Control</li> <li>Class</li> <li>XML File</li> <li>SQL Database</li> <li>Site Map</li> <li>Report</li> <li>Mobile Web Config</li> <li>Browser File</li> </ul>	guration File	
My Templates	<b>s</b> e Templates				
An empty Report I	file for creating a rep	ort using Microsoft reporting technolog	JY		
<u>N</u> ame:	rptPublisher.rdlc				
Language:	Visual C#	□ Pace code □ Select mas	in separate file ter page		
			[	Add	Cancel
cum adaugam	un nou dataso	urce apasand pe Add New I	DataSource		



In ecranul urmator apasam pe NewConnection si selectam baza noastra de date:

Microsoft SQL Server (SqlClient)       Change         Server name:       (local)         (local)       Refresh         Log on to the server       database?         Use Windows Authentication       New Connection         Use SQL Server Authentication       New Connection	
Server name: (local)  Log on to the server  Use Windows Authentication Use SQL Server Authentication User name:	*
(local)	-
Log on to the server     Use Windows Authentication     Use SQL Server Authentication	
Use Windows Authentication     Use SQL Server Authentication	
C Use SQL Server Authentication	
licer namer	In
User Harrier	
Password:	
Save my password	
Connect to a database	
Select or enter a database name:	
Book	
O Attach a database file:	
Browse	
Logical name:	

Putem salva conexiunea in Web.Config(desi o mai avem) si apasam Next. Pe urmatorul ecran lasam selectat "Use SQL Statements" si iarasi Next. Acum scriem codul pentru SQL:

SELECT IDPublisher, NamePublisher, SitePublisher

FROM Publisher

si iarasi Next pina se termina ( sau direct Finish).

Acum in WebSite DataSource ne-a aparut dataset-ul care contine Publisher

Website Data Sources 🚽 🗸 🗸	
询 🛃 📪 🕒	
🖂 🚰 DataSet1	
🖃 🔲 Publisher	
	XV
	R
🚑 Database Exp 🔆 Toolbox 👔 Website Data	

Ignat Andrei

Ne ducem pe Toolbox si tragem pe rptPublisher.rdlc un Table

Ne intoarcem pe WebSite DataSources si tragem Name Publisher pe detaliu:



Acum sa verificam

Cream o noua pagina aspx, frmPublisherReport.aspx si scriem urmatorul cod(sau tragem controlul de

ReportViewer si ii spunem care e raportul)

```
<%@ Page Language="C#" MasterPageFile="~/Book.master" AutoEventWireup="true"
CodeFile="frmPublisherReport.aspx.cs" Inherits="frmPublisherReport"
Title="Report Publisher"
```

```
<%@ Register Assembly="Microsoft.ReportViewer.WebForms, Version=8.0.0.0,
Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a"
```

#### Ignat Andrei

Namespace="Microsoft.Reporting.WebForms" TagPrefix="rsweb" %>

<asp:Content ID="Content1" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">

<rsweb:ReportViewer ID="rptPublisher" runat="server">

<LocalReport ReportPath="rptPublisher.rdlc">

</LocalReport>

</rsweb:ReportViewer>

</asp:Content>

#### In codul C# punem urmatoarele linii :

using System;

using System.Data;

using System.Configuration;

using System.Collections;

using System.Web;

using System.Web.Security;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Web.UI.WebControls.WebParts;

using System.Web.UI.HtmlControls;

using Microsoft.Reporting.WebForms;

public partial class frmPublisherReport : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

BookObjects.ColPublisher publishers = new BookObjects.ColPublisher();

#### Ignat Andrei

publishers.Load();

```
ReportDataSource rds = new
ReportDataSource("DataSet1_Publisher",publishers);
```

rptPublisher.ProcessingMode = ProcessingMode.Local;

rptPublisher.LocalReport.DataSources.Add(rds);

rptPublisher.LocalReport.Refresh();

Si modificam fisierul rdlc deschizindu-l cu notepad-ul si scriind urmatoarele:

<?xml version="1.0" encoding="utf-8"?>

<Report xmlns="http://schemas.microsoft.com/sqlserver/reporting/2005/01/reportdefinition" xmlns:rd="http://schemas.microsoft.com/SQLServer/reporting/reportdesigner">

<DataSources>

}

}

```
<DataSource Name="BookConnectionString">
```

<ConnectionProperties>

<ConnectString />

<DataProvider>SQL</DataProvider>

</ConnectionProperties>

<rd:DataSourceID>c649d533-64c3-42b6-9805-19adbfccd468</rd:DataSourceID>

</DataSource>

</DataSources>

<BottomMargin>1in</BottomMargin>

Ignat Andrei

<RightMargin>1in</RightMargin>

<rd:DrawGrid>true</rd:DrawGrid>

<InteractiveWidth>8.5in</InteractiveWidth>

<rd:SnapToGrid>true</rd:SnapToGrid>

<Body>

<ReportItems>

<Table Name="table1">

<Footer>

<TableRows>

<TableRow>

<TableCells>

<TableCell>

<ReportItems>

<Textbox Name="textbox7">

<rd:DefaultName>textbox7</rd:DefaultName>

<ZIndex>1</ZIndex>

<Style>

<PaddingLeft>2pt</PaddingLeft>

<PaddingBottom>2pt</PaddingBottom>

<PaddingRight>2pt</PaddingRight>

<PaddingTop>2pt</PaddingTop>

</Style>

<CanGrow>true</CanGrow>

<Value />

</Textbox>

</ReportItems>
Tutorial programare in .NET

Ignat Andrei

</TableCell>

</TableCells>

<Height>0.45833in</Height>

</TableRow>

</TableRows>

</Footer>

<Left>0.5in</Left>

<DataSetName>DataSet1\_Publisher</DataSetName>

<Top>0.25in</Top>

<Width>6.5in</Width>

<Details>

<TableRows>

<TableRow>

<TableCells>

<TableCell>

<ReportItems>

<Textbox Name="txtName">

<rd:DefaultName>Name</rd:DefaultName>

<Style>

<PaddingLeft>2pt</PaddingLeft>

<PaddingBottom>2pt</PaddingBottom>

<PaddingRight>2pt</PaddingRight>

<PaddingTop>2pt</PaddingTop>

</Style>

<CanGrow>true</CanGrow>

<Value>=Fields!Name.Value</Value>

Ignat Andrei

</Textbox>

</ReportItems>

</TableCell>

</TableCells>

<Height>0.45833in</Height>

</TableRow>

</TableRows>

</Details>

<Header>

<TableRows>

<TableRow>

<TableCells>

<TableCell>

<ReportItems>

<Textbox Name="textbox1">

<rd:DefaultName>textbox1</rd:DefaultName>

<ZIndex>2</ZIndex>

<Style>

<PaddingLeft>2pt</PaddingLeft>

<PaddingBottom>2pt</PaddingBottom>

<PaddingRight>2pt</PaddingRight>

<PaddingTop>2pt</PaddingTop>

</Style>

<CanGrow>true</CanGrow>

<Value>Name</Value>

</Textbox>

Ignat Andrei

</ReportItems>

</TableCell>

</TableCells>

<Height>0.45833in</Height>

</TableRow>

</TableRows>

</Header>

<TableColumns>

<TableColumn>

<Width>6.5in</Width>

</TableColumn>

</TableColumns>

<Height>1.375in</Height>

</Table>

</ReportItems>

<Height>2in</Height>

</Body>

<rd:ReportID>9f0247a8-15fe-4ef9-962e-c4c670524163</rd:ReportID>

<LeftMargin>1in</LeftMargin>

<DataSets>

<DataSet Name="DataSet1\_Publisher">

<rd:DataSetInfo>

<rd:ObjectDataSourceType>Publisher</rd:ObjectDataSourceType>

<rd:TableName>Publisher</rd:TableName>

</rd:DataSetInfo>

<Query>

Tutorial programare in .NET

Ignat Andrei

<rd:UseGenericDesigner>true</rd:UseGenericDesigner>

<CommandText />

<DataSourceName>BookConnectionString</DataSourceName>

</Query>

<Fields>

<Field Name="IDPublisher">

<rd:TypeName>System.Int32</rd:TypeName>

<DataField>IDPublisher</DataField>

</Field>

<Field Name="Name">

<rd:TypeName>System.String</rd:TypeName>

<DataField>Name</DataField>

</Field>

<Field Name="Site">

<rd:TypeName>System.String</rd:TypeName>

<DataField>Site</DataField>

</Field>

</Fields>

</DataSet>

</DataSets>

<Width>9.75in</Width>

<InteractiveHeight>11in</InteractiveHeight>

<Language>en-US</Language>

<TopMargin>1in</TopMargin>

</Report>

Acum, rulind pagina, observam ca se poate exporta raportul in Excel si PDF – mai mult decit sufficient:

Bepert Publisher Hisrosuff Infe Ole 28 See Fyvortas Dode	rmet Explores	_16 ×
G 644 - () - 🖹 🖻 🐔	💭 Seets 👷 Feaster 🐵 🖉 + 🖕 🖄 + 🛄 🐒 📽 🍪	
Agliters 📳 Http://localhoet/Book/WebyT	hmPLbbleferSapat.app. ddobaannan oo daalaa	· · · · · · · · · · · · · · · · · · ·
	Book appäration English 📰	
	11 1 1 11 11 11 110% E Fut these	
	Selent a format 🔄 format 🔮	
	Aurobat (PDP) file	
	Name	
	Andrei	
= bdan All publishers		
All books		
	4	
Daw		Local etranet
	1 CV	
Acoloci roport co	a postavadas si su COL Canvar Departing Canvios	
Acerasi raport se	2 poale vedea si cu SQL Server Reporting Services	
inkuri utile:		
http://www.goti	reportviewer.com/ - in partea dreapta aveti exemple de aplicatii	care fac diverse –
genereaza autor	mat fisierele rdlc, scot automat fisierele Excel si multe altele	
-		
SQL Server Repo	orting Services – pentru utilizarea avansata a rapoartelor.	

## Programarea in .NET - partea a 17-a Rapoarte in ASP.NET –Windows Forms

Haideti sa repetam ceea ce am facut in ASP.NET pentru WindowsForms

Deschideti Book.sln, adaugati o noua forma in proiect(frmPublisherPrint.cs) si trageti ReportViewer in forma.(Daca nu il gasiti, click dreapta in Toolbox, alegeti "choose items", cautati ReportViewer din namespace-ul Microsoft.Reporting.WinForms si selectati-l

Ch	Choose Toolbox Items							
ŀ	.NET Framework Components COM Components							
	Name	Namespace	Assembly Name	Directory				
	ReportClientDocumen	CrystalDecisions.ReportAppSer	CrystalDecisions.Cryst	Global Ass				
	ReportDocument	CrystalDecisions.CrystalReport	CrystalDecisions.Cryst	Global Ass				
	ReportExecutionService	Microsoft.SqlServer.ReportingS	Microsoft.ReportViewer	Global Ass				
	ReportExporter	CrystalDecisions.Reporting.We	CrystalDecisions.Web (	Global Ass				
	ReportServiceBase	CrystalDecisions.Web.Services	CrystalDecisions.Web (	Global Ass				
	ReportViewer	Microsoft.Reporting.WebForms	Microsoft.ReportViewer	Global Ass				
	ReportViewer	Microsoft.Reporting.WinForms	Microsoft.ReportViewer	Global Ass				
	RequiredFieldValidator	System.Web.UI.MobileControls	System.Web.Mobile (2	Global Ass				
	RequiredFieldValidator	System.Web.UI.WebControls	System.Web (2.0.0.0)	Global Ass 👻				
	•							
	Filter:			Clear				
	ReportViewer							
	🔜 Language: Invaria	ant Language (Invariant Country)		Browse				
	Version: 8.0.0.	0						
			OK Cancel	Reset				

Adaugam raportul existent prin click dreapta pe BookWin



Va duceti in BookWeb, alegeti din casuta "Files of type" ultima selectie "All files" si selectati rptPublisher.rdlc.

Acum click pe el si in fereastra de proprietati alegeti la "Copy to output directory " "Copy always"

Properties	- ₽ X
rptPublisher.rdlc File Properties	•
Build Action	None
Copy to Output Directory	Copy always 💽
Custom Tool	
Custom Tool Namespace	
File Name	rptPublisher.rdlc
Full Path	C:\Book\BookWin\rptPublisher.rdlc

}

Bun – acum au ramas 3 lucruri de facut : vizualizarea formei ca actiune, legarea controlului de raportul existent si codul de incarcare a datelor in raport.

Pentru vizualizarea formei ca actiune adaugati un buton btnPrint in frmPublisherList iar pe eveniment scrieti urmatorul cod:

```
private void btnPrint_Click(object sender, EventArgs e,
{
    frmPublisherPrint p = new frmPublisherPrint();
    p.ShowDialog(this);
```

Pentru legarea controlului vom seta la proprietati calea catre raport(presupunem ca se va afla in acelasi folder) si processing mode la local

Properties 🗸 🗸 🗸				
rptPublisher Microsoft.Reporting.Wi	•			
₽∎ ≵↓   🗉 🖋   🖻				
🗆 LocalReport		<b></b>		
DisplayName				
EnableExternalImages	False			
EnableHyperlinks	False			
ReportEmbeddedResource				
ReportPath	rptPublisher.rdlc			
ProcessingMode	Local	-		
E ServerReport     ServerReport				
🗆 Toolbar	Toolbar			
ShowBackButton	True			
ShowDocumentMapButton	True			
ShowExportButton	True	-		

```
Tutorial programare in .NET
                                                                            Ignat Andrei
Ultimul lucru de facut – incarcarea colectiei pe evenimentul de load :
rivate void frmPublisherPrint Load(object sender, EventArgs e)
         {
             BookObjects.ColPublisher publishers = new
BookObjects.ColPublisher();
             publishers.Load();
             MessageBox.Show(""+publishers.Count);
             ReportDataSource rds = new ReportDataSource ("DataSet
publishers);
             rptPublisher.ProcessingMode = ProcessingMode.Local;
             rptPublisher.LocalReport.DataSources.Clear()
             rptPublisher.LocalReport.DataSources.Add(rds
             rptPublisher.LocalReport.Refresh()
             rptPublisher.RefreshReport()
         }
( exact acelasi cod ca la Web, in afara liniei :
```

Ea previne cazul( desintilnit) in care editorul IDE adauga, cu de la sine putere, un ReportDataSource

Ceea ce se va infatisa va fi:



## Tutorial programare in .NET



Ce mai e de facut

1)frmPublisherPrint sa nu mai afiseze ce vrea ea - ci sa primeasca un argument(in constructor, de exemplu) care sa spun ce lista de publisher-i are de afisat

2) Avind in vedere ca rapoartele sunt aceleasi pentru Windows si Web, ar fi interesant de facut un dll care sa intoarca raport – ul cerut