

Programarea Aplicațiilor Windows – curs 11

Conf. dr. Cristian CIUREA
Departamentul de Informatică și Cibernetică Economică
Academia de Studii Economice București
cristian.ciurea@ie.ase.ro

Agenda

1. Lucrul cu baze de date (partea a II-a)
2. Reporting

Lucrul cu baze de date

DataSet – container de date, care poate conține mai multe obiecte **DataTable**, câte unul pentru fiecare tabelă din baza de date.

```
DataSet dataSet = new DataSet();  
DataTable tabela =  
dataSet.Tables["student"];
```

Lucrul cu baze de date

DataAdapter – mediază schimburile de date dintre un **DataSet** și baza de date, iar principala metodă, *Fill(dataSet, tabela)*, încarcă cu date o tabelă dintr-un DataSet.

```
OleDbDataAdapter adaptor = new  
OleDbDataAdapter("SELECT * FROM student",  
conexiune);  
DataSet dataSet = new DataSet();  
adaptor.Fill(dataSet, "student");
```

Lucrul cu baze de date

Fiecare **DataAdapter** mediază transferul de date între un singur obiect **DataTable** din **DataSet** și rezultatul unei singure interogări printr-o comandă SQL.

Lucrul cu baze de date

DataView – oferă un mod de vizualizare a unui obiect **DataSet**, oferind posibilitatea filtrării datelor dintr-o tabelă. Aceleași tabele ale unui DataSet pot avea mai multe vizualizări asociate, reprezentând puncte de vedere diferite ale unor utilizatori.

```
DataTable tabela =  
    dataSet.Tables["student"] ;  
DataView dv = new DataView(tabela) ;  
dv.Sort = "nume" ;  
dv.RowFilter = "forma='ZI' " ;
```

Entity Framework

Entity Framework (EF) este un framework open-source de mapare obiect-relațională (ORM) pentru ADO.NET.

EF a fost parte a .NET Framework, dar începând cu versiunea 6 (Oct. 2013) s-a separat de .NET Framework.

Entity Framework este un set de tehnologii în ADO.NET care sprijină dezvoltarea de aplicații software orientate spre date.

Entity Framework

O instanță a clasei **DbContext** reprezintă o combinație de modele și unități de lucru, care să poată fi utilizată pentru a interoga dintr-o bază de date și pentru a grupa modificările care vor fi scrise din nou în baza de date ca o unitate de lucru.

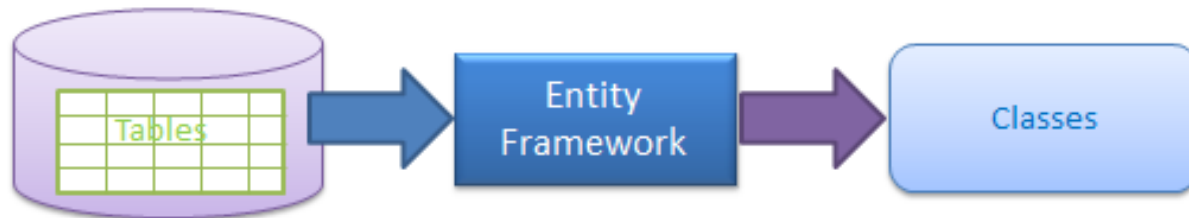
Clasa **DbContext** este utilizată de regulă cu un tip derivat care conține proprietățile **DbSet<TEntity>** pentru entitățile rădăcină ale modelului. Aceste seturi sunt inițializate automat atunci când instanța clasei derivate este creată.

Entity Framework

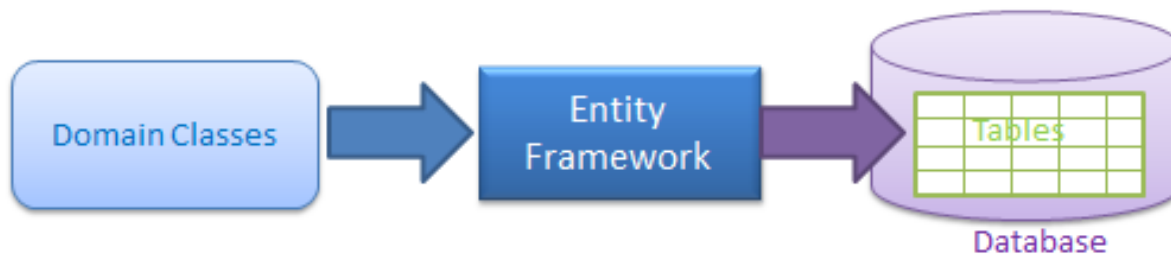
Prin convenție, clasa **DbContext** creează o bază de date pe baza următoarelor criterii:

- dacă este disponibilă o instanță locală de SQL Express, atunci baza de date va fi creată pe acea instanță;
- dacă instanța SQL Express nu este disponibilă, atunci LocalDb va fi utilizată dacă aceasta este disponibilă;
- dacă ambele sunt disponibile, atunci se va utiliza SQL Express;
- baza de date este denumită pe baza namespace-ului, la care se adaugă denumirea clasei derivate din DbContext.

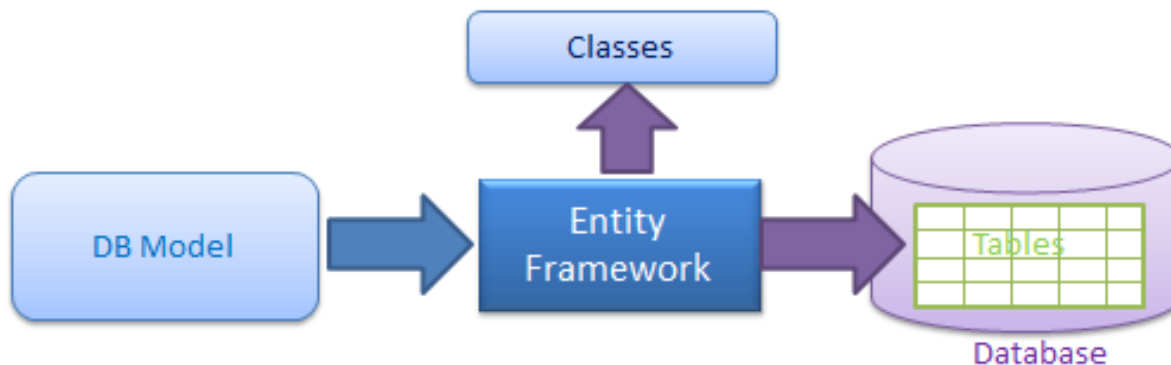
Entity Framework



Generate Data Access Classes for Existing Database



Create Database from the Domain Classes



Create Database and Classes from the DB Model design

Reporting

Pentru realizarea de rapoarte în aplicațiile Windows se poate utiliza controlul **ReportViewer**, disponibil începând cu versiunea .NET Framework 4.0.

Reporting

Form3

1 of 1 | 100% | Find | Next

cod	nume	varsta	inaltime	CNP	sex	forma
222	Popescu	79	175	1850415020058	F	ID
223	Gigel	25	187	123456778	M	ID
224	Dorel	23	157	1234567890	M	ZI
225	Georgel	25	178	123456789	M	ZI

Bibliografie

- [1] I. Smeureanu, M. Dârdală, A. Reveiu – *Visual C# .NET*, Editura CISON, București, 2004.
- [2] C. Petzold – *Programming Microsoft Windows with C#*, Microsoft Press, 2002.
- [3] L. O'Brien, B. Eckel – *Thinking in C#*, Prentice Hall.
- [4] J. Richter – *Applied Microsoft .NET Framework Programming*, Microsoft Press, 2002.
- [5] <http://acs.ase.ro/paw>