

Laborator 4

ORM – Studiu de caz: Entity Framework

PARTEA 1 – Crearea bazei de date cu Microsoft SQL Server

1. Creați baza de date.
 - a. Click pe *Databases – New Database...* (vezi Figura 1)

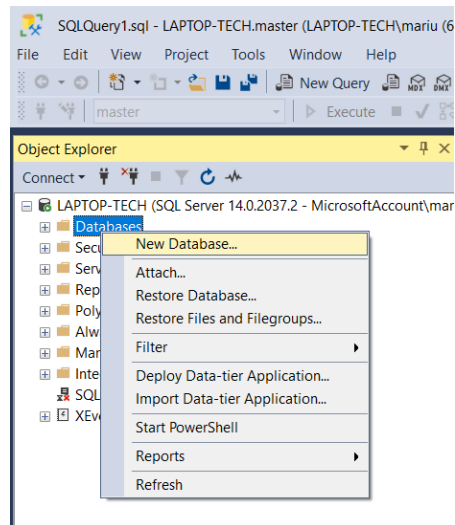


Figura 1. Modalitatea de creare a unei baze de date

- b. La *Database name* introduceți numele bazei de date, exact ca în Figura 2. Click *Ok*.

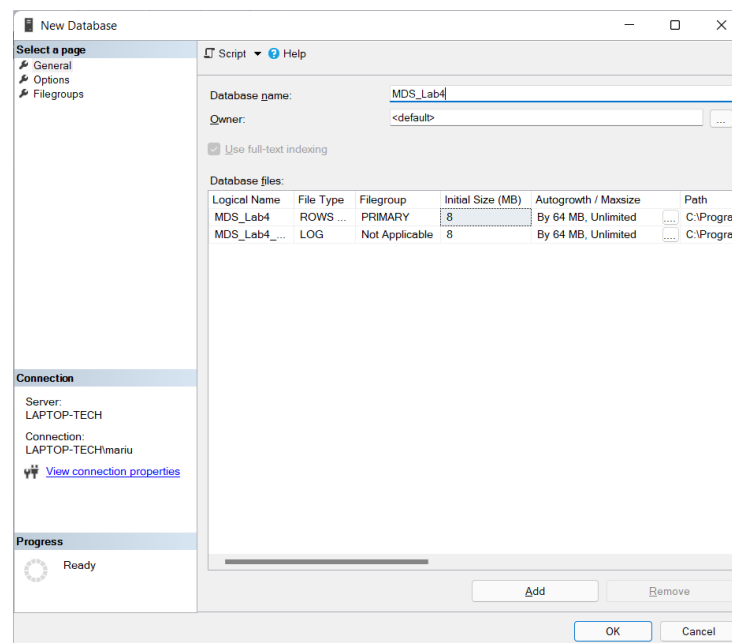


Figura 2. Introducerea numelui bazei de date

2. Verificați că baza de date a fost creată cu succes.

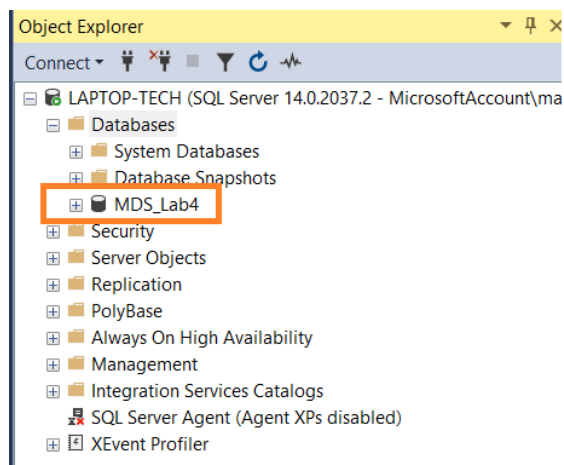


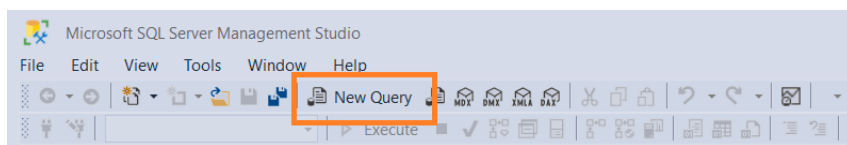
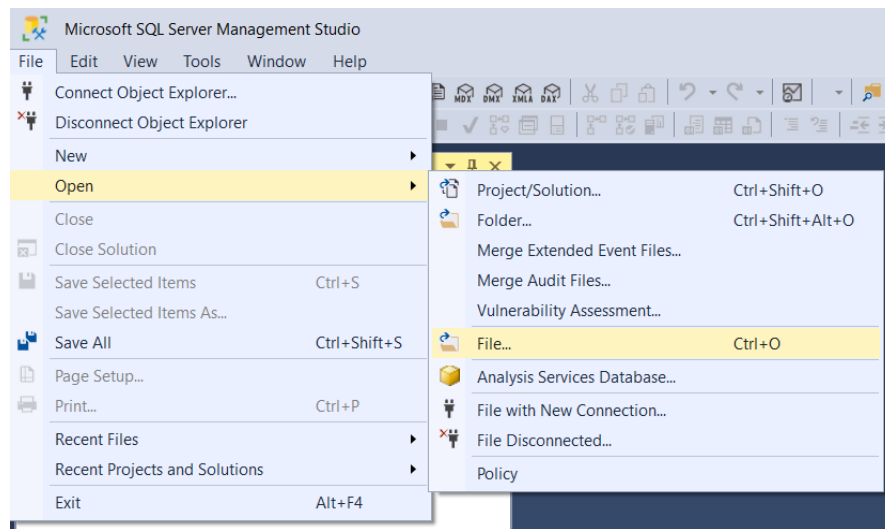
Figura 3. Verificarea creării bazei de date

3. Rulați cele două scripturi *.sql, în ordinea menționată mai jos

a. *query_create_tables.sql*

b. *query_insert_date.sql*

Pentru a rula cele două script-uri, deschideți Microsoft SQL Server Management Studio, pe bara de sub meniu, clic pe *New Query* (vezi Figura 4) sau *File – Open – File...* (vezi Figura 5).

Figura 4. Opțiunea *New Query*Figura 5. Opțiunea *File - Open – File...*

4. Rulați fiecare script cu F5 sau click pe butonul *Execute* (vezi Figura 6)

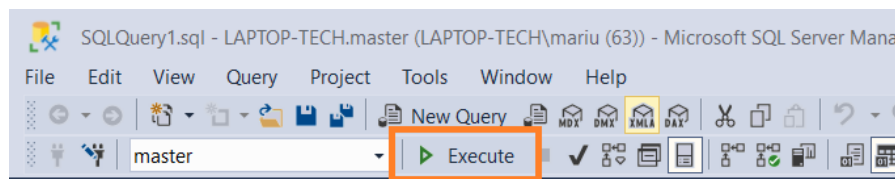


Figura 6. Rularea scripturilor

5. Verificați că tabele au fost create (vezi Figura 7) și că există datele introduse corespunzător pentru tabelul *Articole* (vezi Figura 8) și tabelul *Familie* (vezi Figura 9).

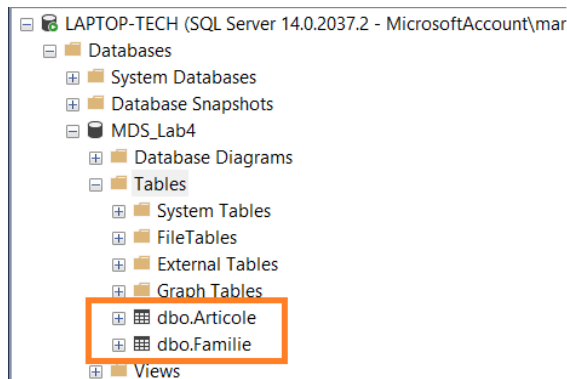


Figura 7. Tabele create în urma rulării scripturilor

LAPTOP-TECH.MDS_Lab4 - dbo.Articole				
	IDArticol	CodArticol	DescriereA	CodArticol...
▶	1	A01	Articol Test 1	F01
	2	A02	Alt articol t...	F01
	3	A03	A. 03	F02
	4	A04	A. 04	F02
	5	A05	A. 05	F02
	6	A06	A. 06	F02
	7	A07	A. 07	F03
	8	A08	A. 08	F04
*	NULL	NULL	NULL	NULL

Figura 8. Tabelul *Articole*

LAPTOP-TECH.MDS_Lab4 - dbo.Familie	
	DescriereFamilie
▶	F01
	F02
	F03
	F04
*	NULL

Figura 9. Figura *Familie*

Partea 2 – Crearea aplicației în Visual Studio

1. Creați o aplicație bazată pe WinForms (vezi Figura 10). După selectarea WinForms, click *Next*.

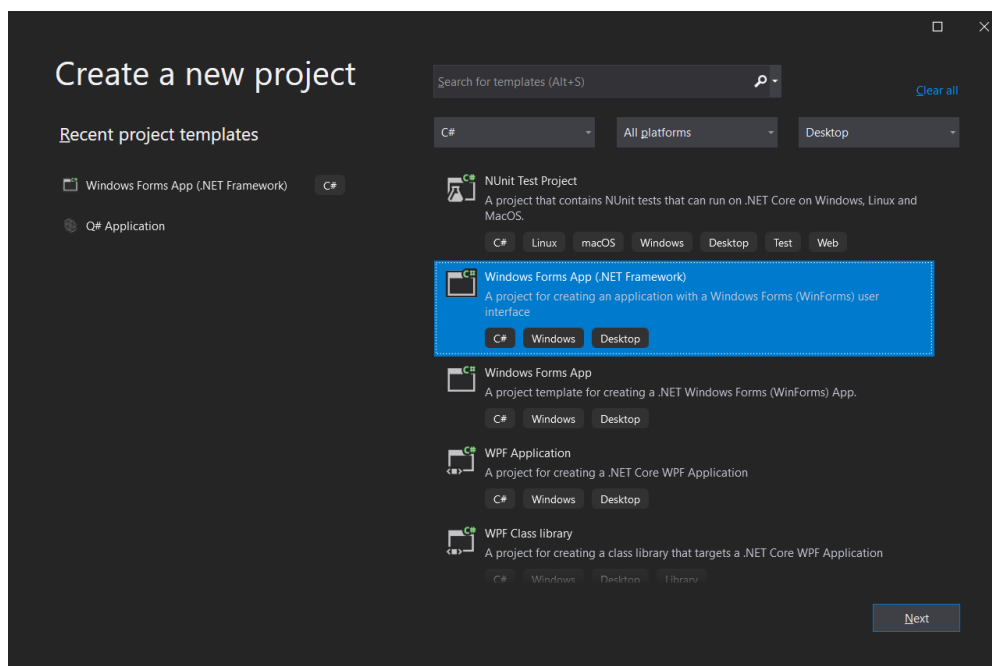


Figura 10. Crearea unei aplicații bazată pe WinForms

2. Introduceți numele aplicației, *MDS_Exemplu4_EntityFramework*.

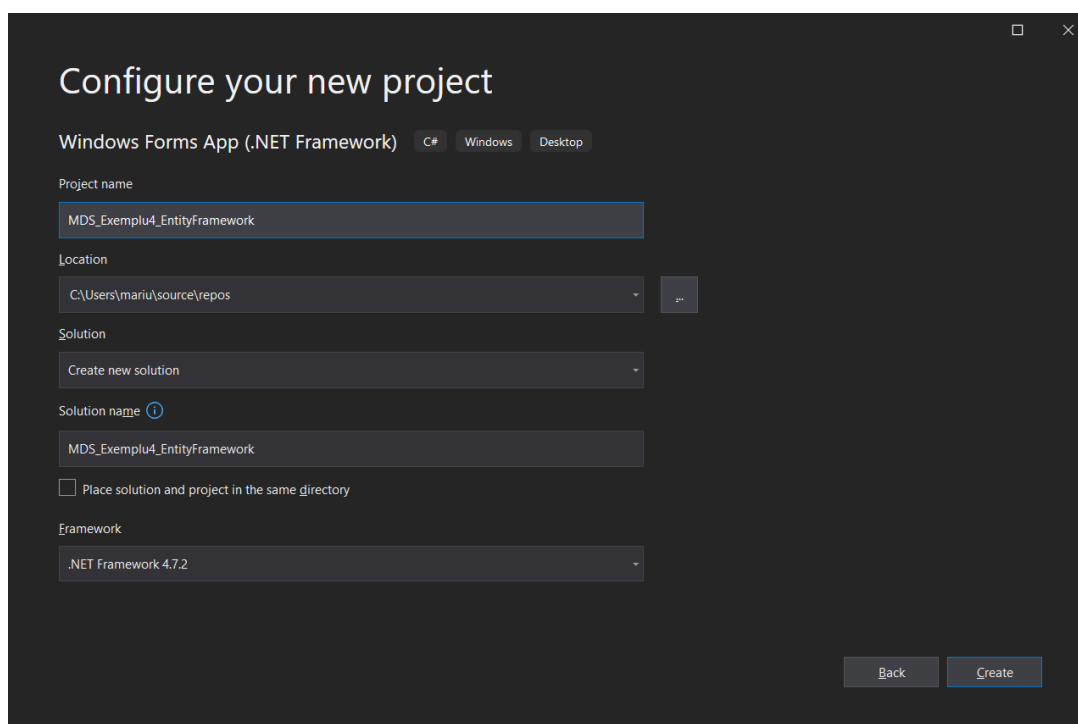


Figura 11. Introducerea numelui aplicației.

3. Următoarea fereastră ar trebui afișată

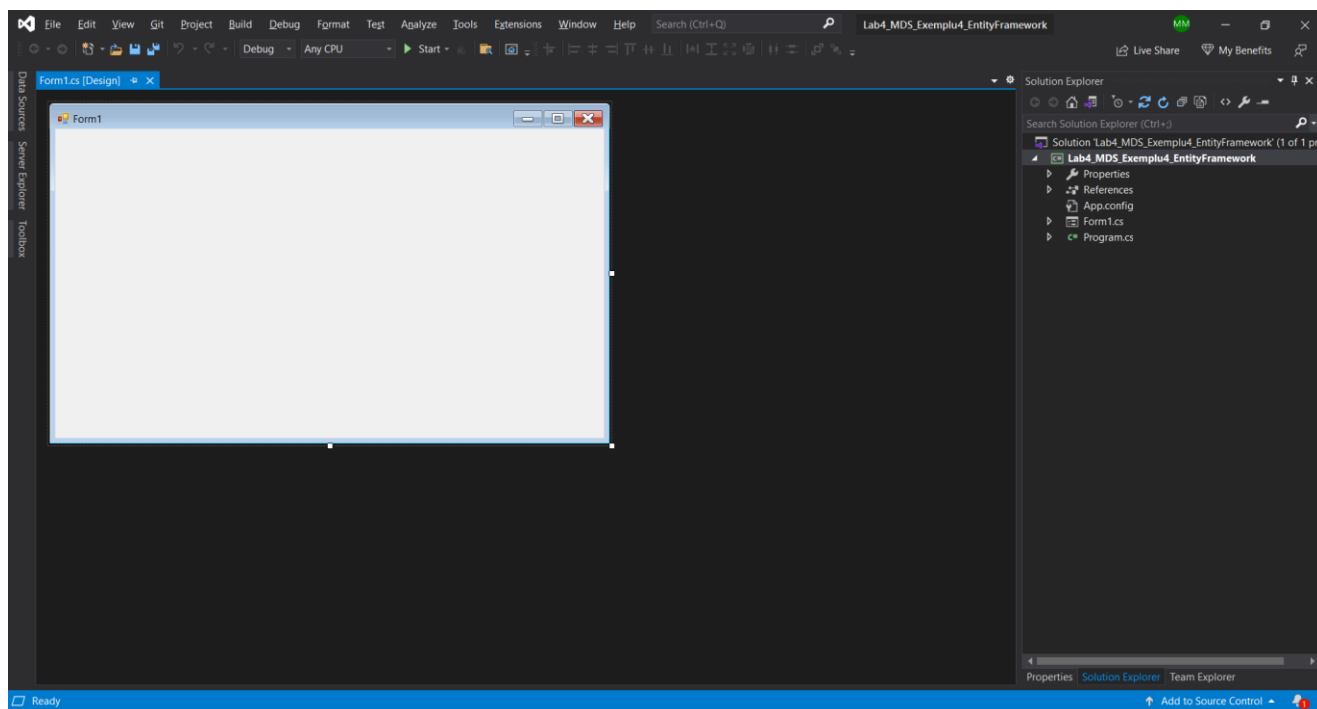


Figura 12. Fereastra aplicației nou create

4. Modelarea claselor cu EF. Apăsați **Ctrl+Shift+A** și selectați **Data** și apoi **ADO.NET Entity Data Model** (vezi Figura 13). Puteți să lăsați același nume din textbox-ul din partea de jos a ecranului, **Model1**. Apoi click pe butonul **Add**.

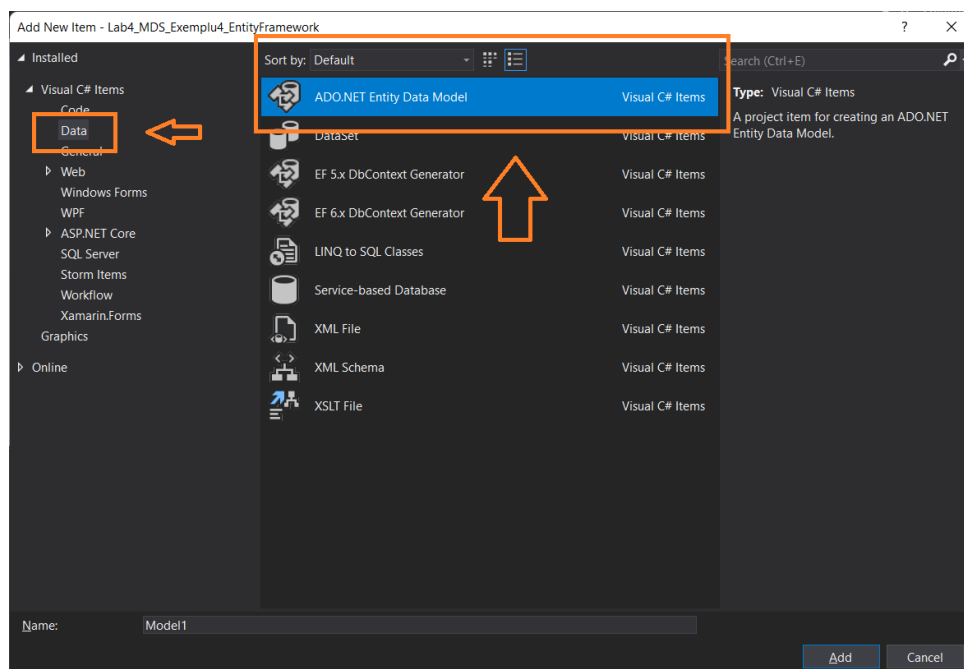


Figura 13. Modelarea claselor

5. Selectați **EF Designer from Database** și click **Next**

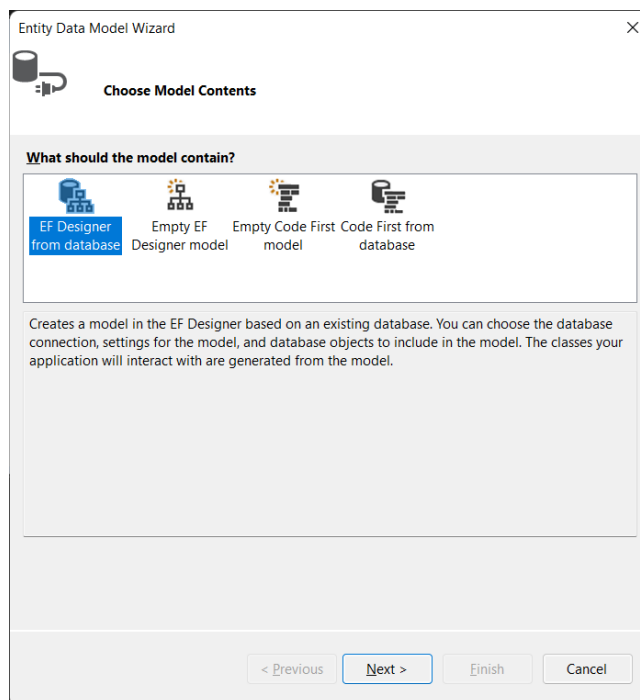


Figura 14. EF Designer from Database

6. Selectați baza de date creată anterior și lăsați totul implicit. Click *Next*.

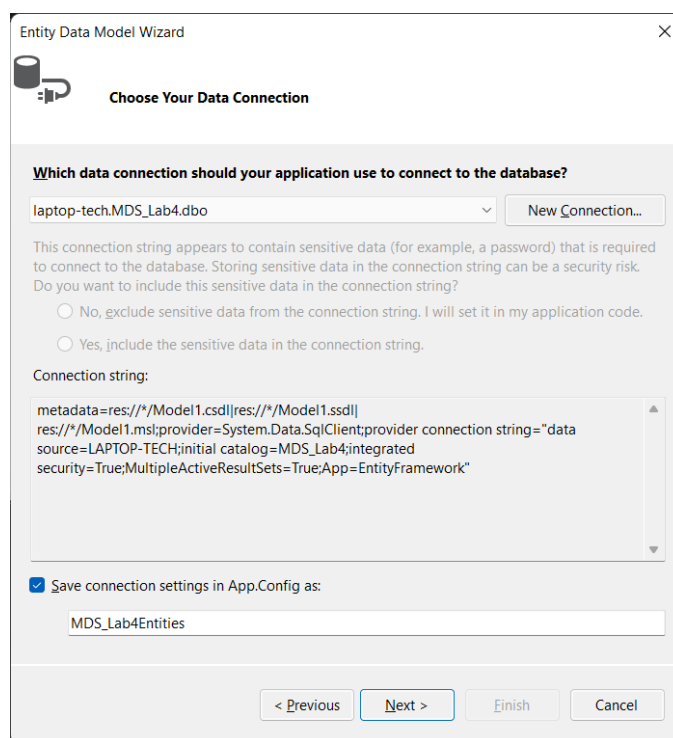


Figura 15. Entity Data Model Wizard

7. Selectați *Entity Framework 6.x* și click pe *Next*.

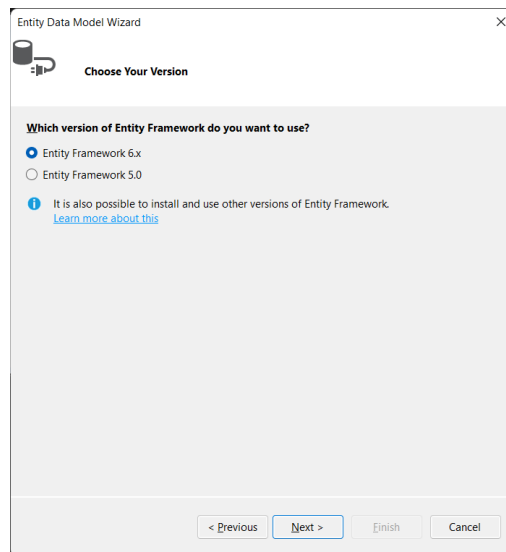


Figura 16. Entity Data Model Wizard

8. Selectați exact cum se vede în imagine. Click *Finish*.

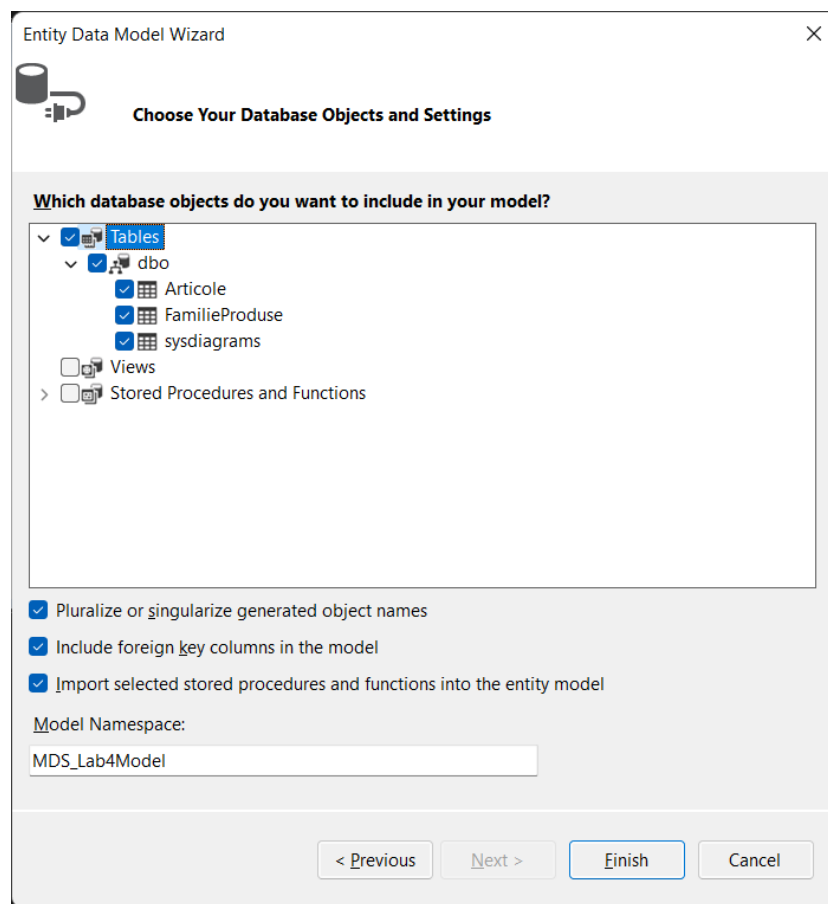


Figura 17. Entity Data Model Wizard

9. Dacă primiți mesajul din figura de mai jos, bifați checkbox-ul aferent și click Ok.

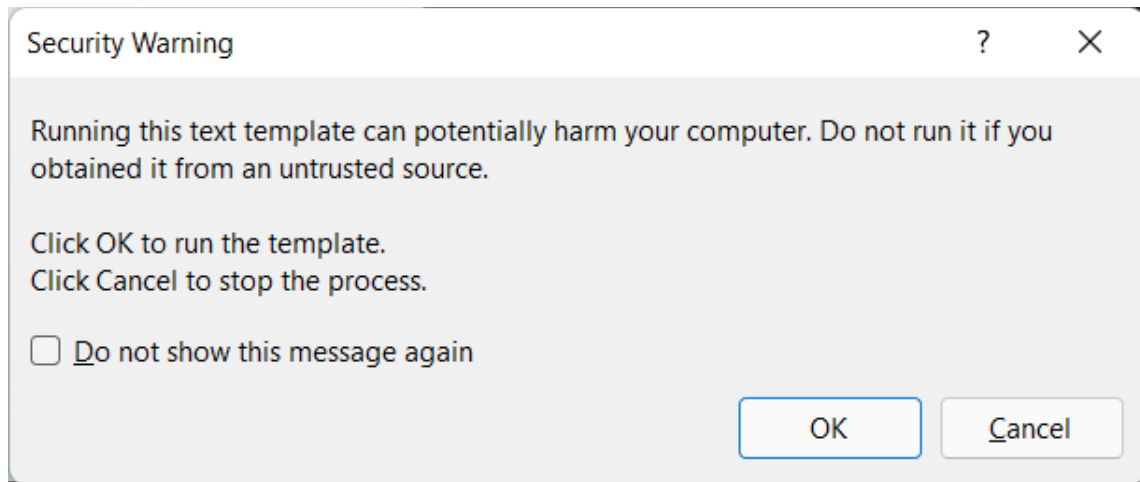


Figura 18. Security Warning

10. Verificați modelul *Model1.edmx*, cele două module *EntityFramework* și *EntityFramework.SqlServer*

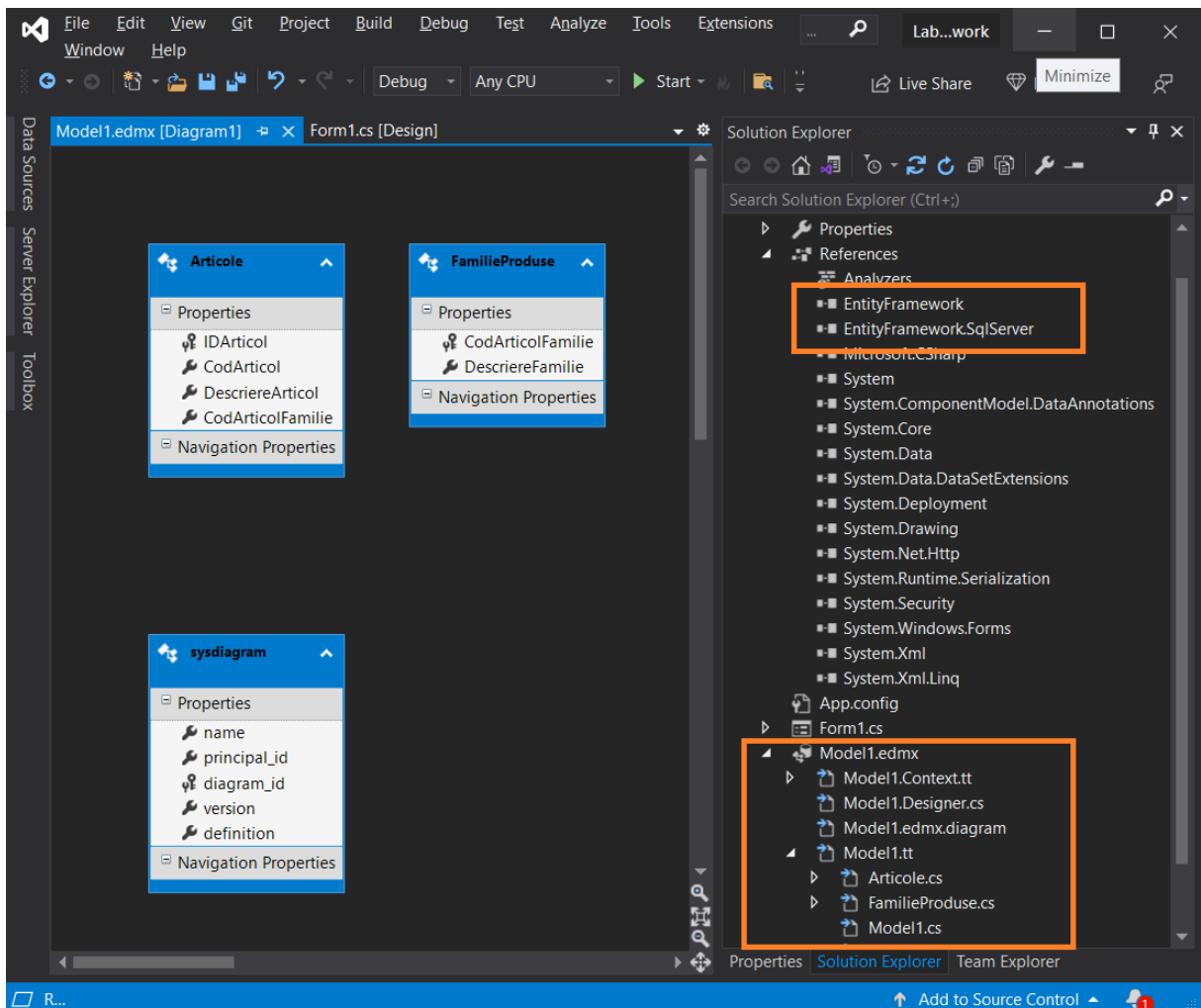


Figura 19. Modelarea claselor

11. Creați o simplă interogare (query) prin urmarea următorilor pași:

11.1. (În cazul Visual Studio instalat și configurat pe calculatorul meu), accesați din dreapta Toolbox.

Dacă acesta nu există, apăsați *Ctrl + Alt + X* sau mergeți în meniul *View – Toolbox*.

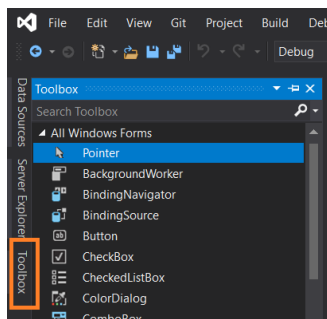


Figura 20. Accesarea Toolbox-ului

11.2. Selectați controlul *Button* și trageți acesta pe formular (fereastră).

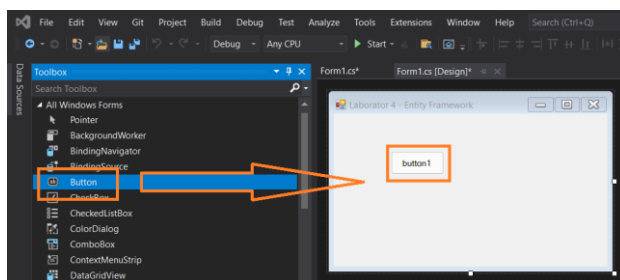


Figura 21. Amplasarea butonului pe formular

11.3. Click dreapta pe butonul amplasat pe fereastră și click pe *Properties*.

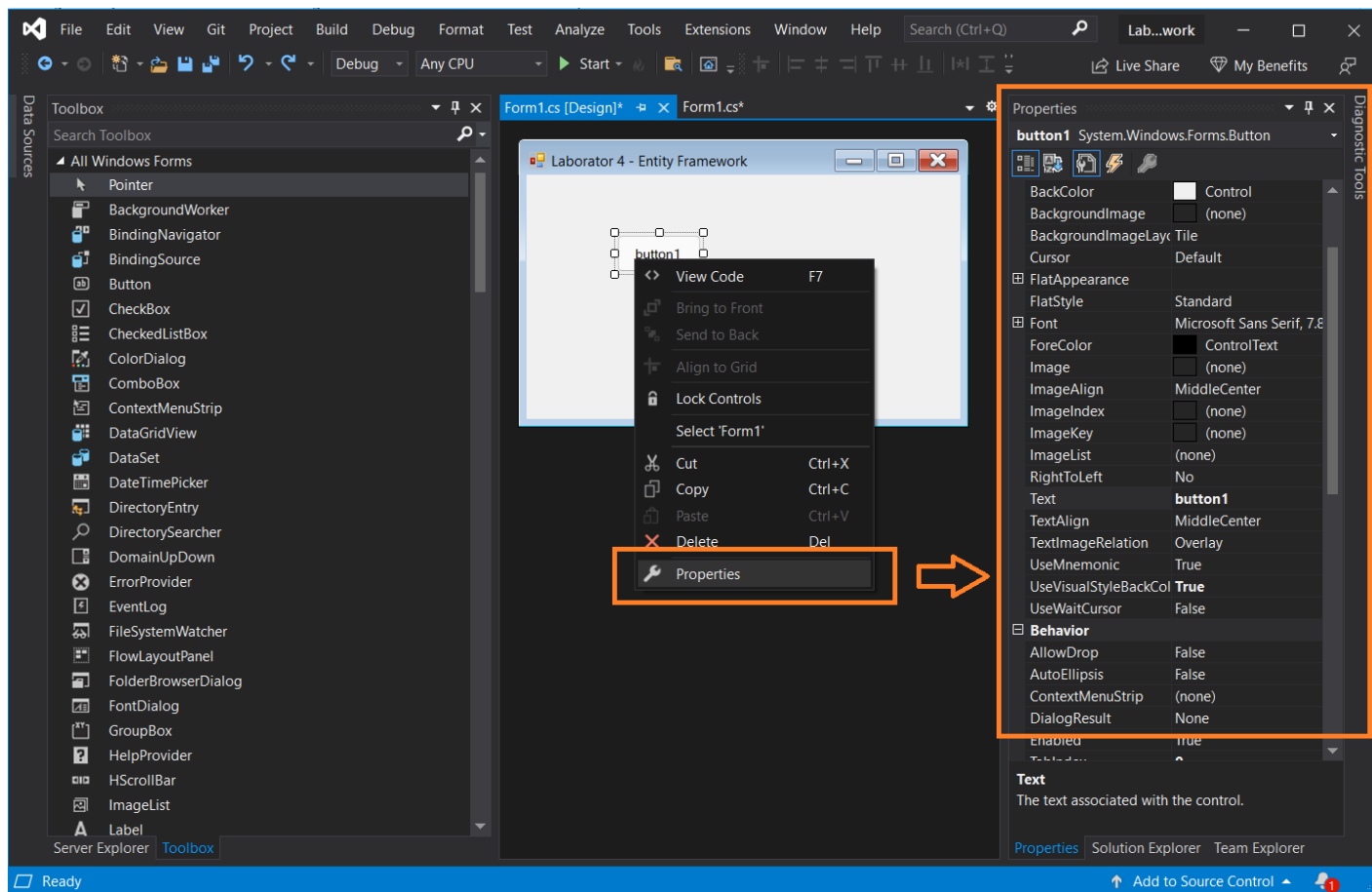


Figura 22. Deschiderea tab-ului (secțiunii) cu proprietăți pentru butonul nou amplasat pe formular

- 11.4. În fereastra cu proprietăți (*Properties*) căutați proprietatea *Text* și (*Name*) și redenumiți acestea corespunzător, utilizând un nume cât mai sugestiv. Exemplu: Pentru proprietatea *Text* introduceți valoarea *Afiseaza Articol* și pentru proprietatea (*Name*) introduceți valoarea *btnAfiseazaArticol*. După introducerea valorilor respective, butonul ar trebui să arate ca în Figura 23, proprietatea (*Name*) ca în Figura 24, și proprietatea *Text* ca în Figura 25.

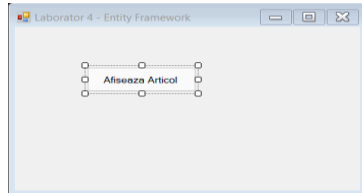
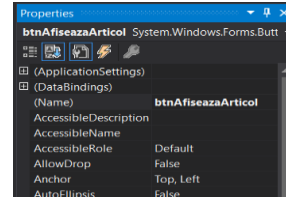
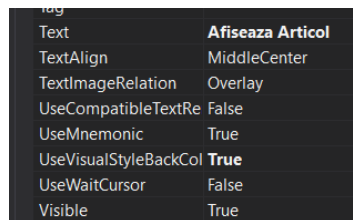


Figura 23. Butonul cu noul nume adăugat

Figura 24. Proprietatea (*Name*)Figura 25. Proprietatea *Text*

- 11.5. Pentru adăugarea și realizarea interogării, faceți dublu click pe butonul de pe formular, și în funcția generată (intitulată *btnAfiseazaArticol*) adăugați codul ca în imaginea alăturată. **Atenție: Exemplul vă pune la dispoziție un exemplu care utilizează valori hard-coded. Nu este o practică benefică. Acest mod de programare nu se folosește în realitate și trebuie evitat în totalitate, astfel încât să nu genereze breșe de securitate și alte dezavantaje care privesc integritatea aplicației.**

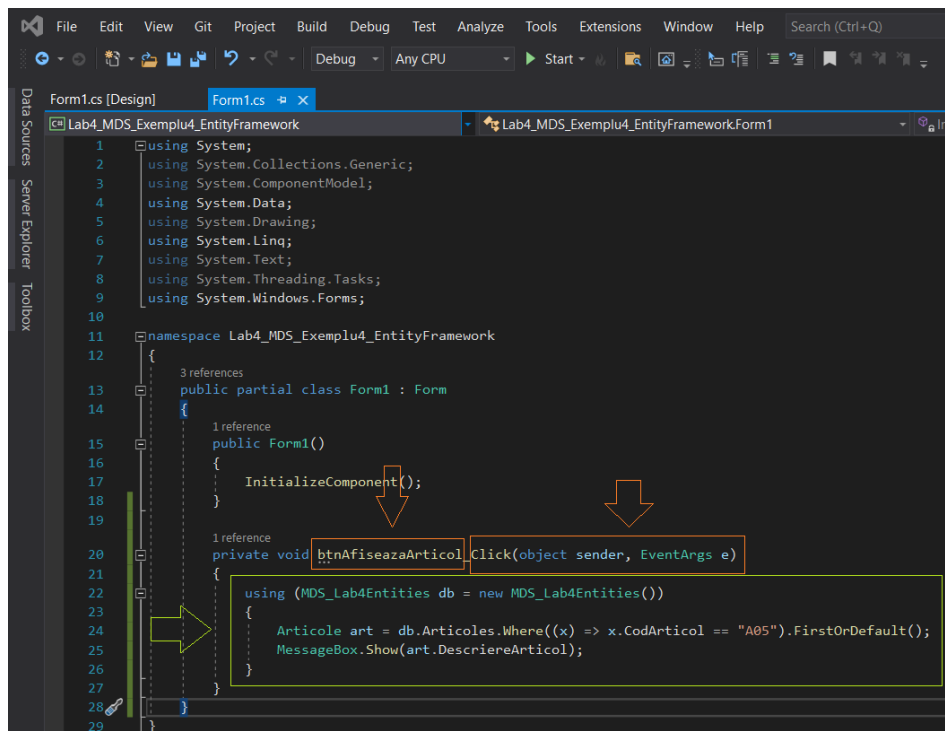


Figura 26. Exemplu de interogare

- 11.6. După ce ați examinat codul suficient și ați înțeles ce se întâmplă, rulați exemplul prin F5 sau apăsați butonul *Start* din bară, cel cu săgeata verde. Codul de la pasul 11.5 nu face altceva decât să returneze descrierea articolului pentru A05.

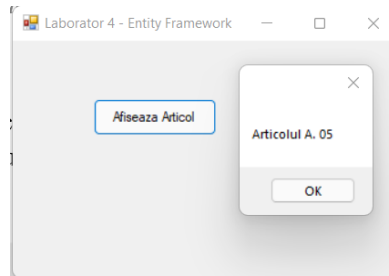
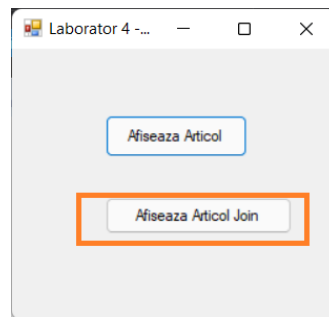


Figura 27.

- 11.7. După cum probabil ați reușit să intuiți, avem două tabele în baza de date sau entități la nivel de aplicație, intitulate *Articole* și *FamilieProduce* (acesta se referă la familia de produse/articole). Acestea sunt legate între ele. Până acum nu am definit cheile străine (foreign key) care să ne ajute și să spună EF-ului să lege cele două tabele între ele, tabele care în momentul de față sunt independente. În condițiile actuale, trebuie să implementăm instrucțiuni mai complexe, care să implementeze *join-uri*, care lipsesc la nivelul modelului. Să presupunem că dorim să selectăm înregistrarea A05 din tabelul *Articole* și în același timp dorim să afișăm descrierea produsului. Pentru a face acest lucru, creați un alt buton intitulat *Afișează Articol (Join)* cu următoarele proprietăți, *Text* are valoarea *Afișează Articol* și *(Name)* are valoarea *btnAfișeazăArticolJoin*. Noul buton arată ca în Figura 28.

Figura 28. Noul buton cu *Afișare Articol* utilizand interogari complexe

- 11.8. Adăugați codul ca în Figura 29.

```
1 reference
private void btnAfișeazăArticolJoin_Click(object sender, EventArgs e)
{
    using (MDS_Lab4Entities db = new MDS_Lab4Entities())
    {
        var articolProduce = db.Articole
            .Join(db.FamilieProduce,
                articol => articol.CodArticolFamilie,
                familieProduce => familieProduce.CodArticolFamilie,
                (articol, familieProduce) => new { Articol = articol, FamilieProduce = familieProduce })
            .Where((x) => x.Articol.CodArticol == "A05")
            .FirstOrDefault();

        MessageBox.Show(articolProduce.Articol.DescriereArticol + " - " + articolProduce.FamilieProduce.DescriereFamilie);
    }
}
```

Figura 29. Codul pentru realizarea interogării complexe, prin realizarea de join.

11.9. Rulați și click pe noul buton. Rezultatul ar trebui să fie ca în Figura 30.

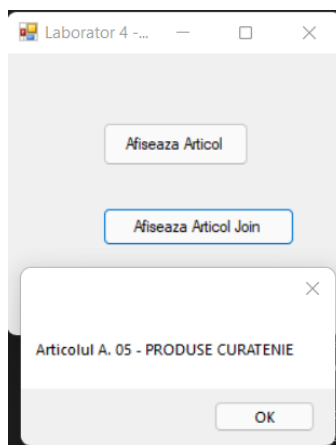


Figura 30. Rezultatul afișat în urma operației de *join*

12. Adăugați o cheie străină (foreign key) pentru cele două tabele folosind Microsoft SQL Server Studio.

12.1. Mergeți în SQL Server Management Studio.

12.2. Deschideți tabelul *Articole* și navigați la *Keys*.

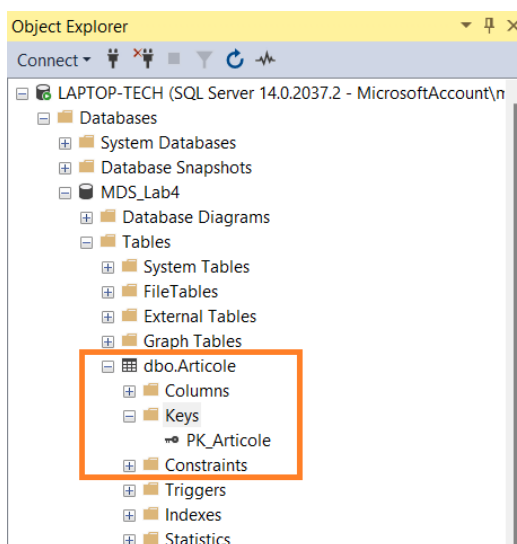


Figura 31. Vizualizarea cheilor

12.3. Clic dreapta pe *Keys* și selectați *Foreign Key*.

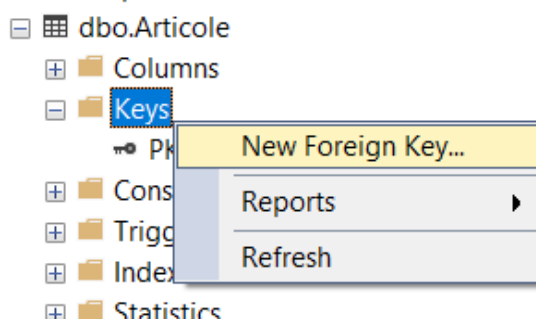


Figura 32. Adăugarea unei chei străine

12.4.Deschideți nodul *Tables and Columns Spec* făcând click pe butonul cu cele trei puncte din dreptul secțiunii respective (vezi Figura 33). Selectați cu atenție câmpurile după următoarea schemă și verificați la final că totul arată ca în Figura 34.

- Foreign Key Base Table = Articole
- Foreign Key Columns = CodArticolFamilie
- Primary/Unique Key Base Table = FamilieProduce
- Primary/Unique Key Columns = CodArticolFamilie

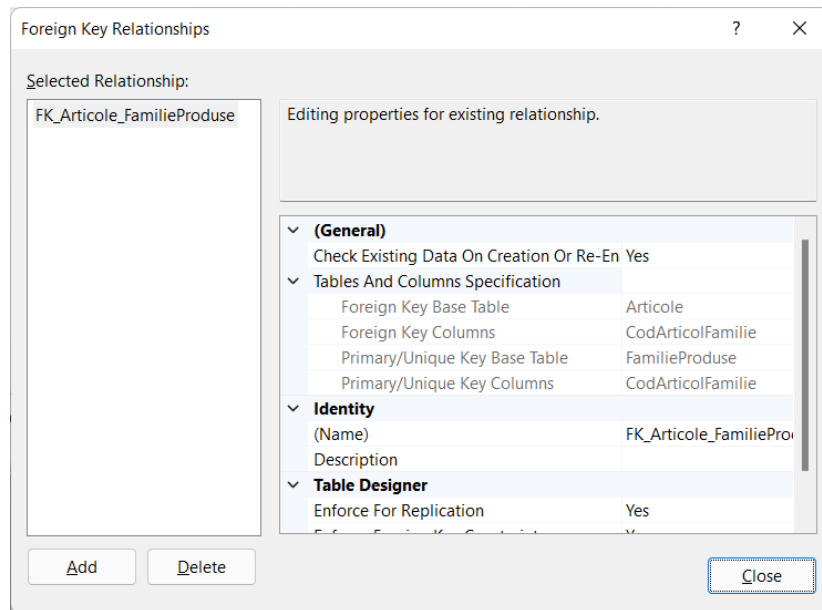


Figura 33. Configurarea cheii străine

12.5.Mergeți în Visual Studio și identificați în *Solution Explorer* fișierul *Model.edmx*. Dublu click pentru deschidere și apoi click dreapta oriunde ca în Figura 34. Selectați *Update model from database...*

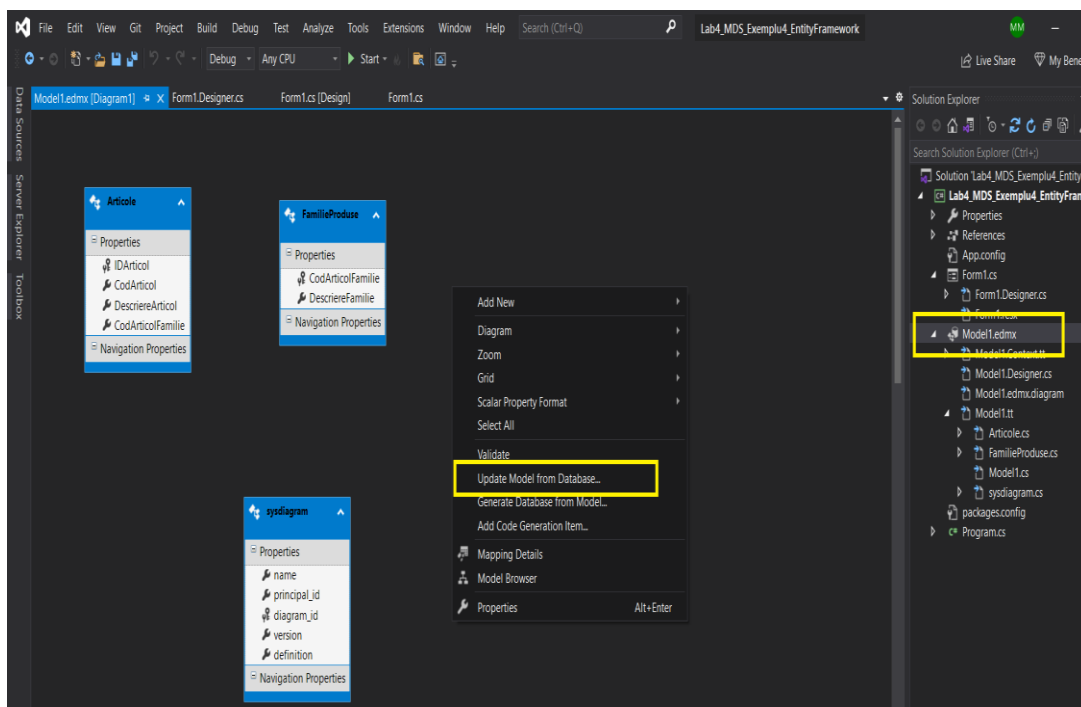


Figura 34. Actualizarea relațiilor și tabelelor

12.6. Click pe *Finish* în *Update Wizard*. Lăsați totul așa cum este.

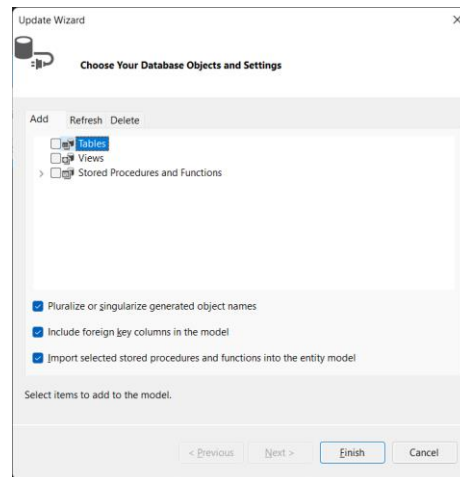


Figura 35. Update Wizard

12.7. Verificați că relația este afișată corespunzător între tabelele respective.

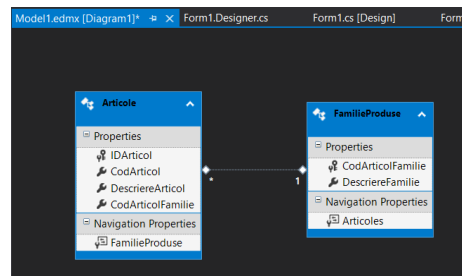


Figura 36. Relația pentru cheie străină dintre cele două tabele

12.8. Adăugați următoarele linii de cod în fișierul *Articole.cs* și *FamilieProduce.cs* din cadrul *Model1.tt* din *Model1.edmx*.

Fișierul *Articole.cs* - adăugați linia de cod de mai jos, astfel încât să arate ca în imaginea alăturată.

```
public virtual FamilieProduce FamilieProduce { get; set; }
```

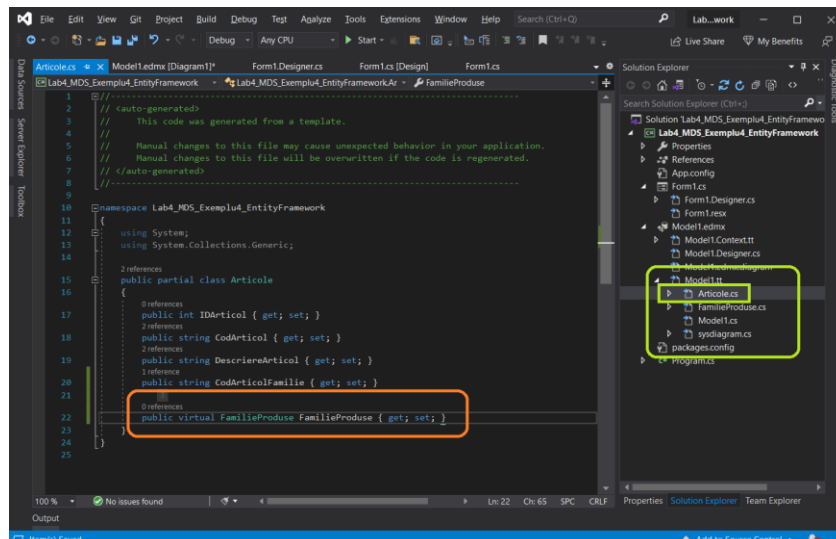


Figura 37. Modificarea este marcată în chenar portocaliu

Fișierul *FamilieProduce* - adăugați liniile de cod de mai jos, astfel încât să arate ca în imaginea alăturată.

- Pentru constructor

```
public FamilieProduce()
{
    this.Articole = new HashSet<Articole>();
}
```

- Linia de cod pentru colecția articolelor

```
public virtual ICollection<Articole> Articole { get; set; }
```

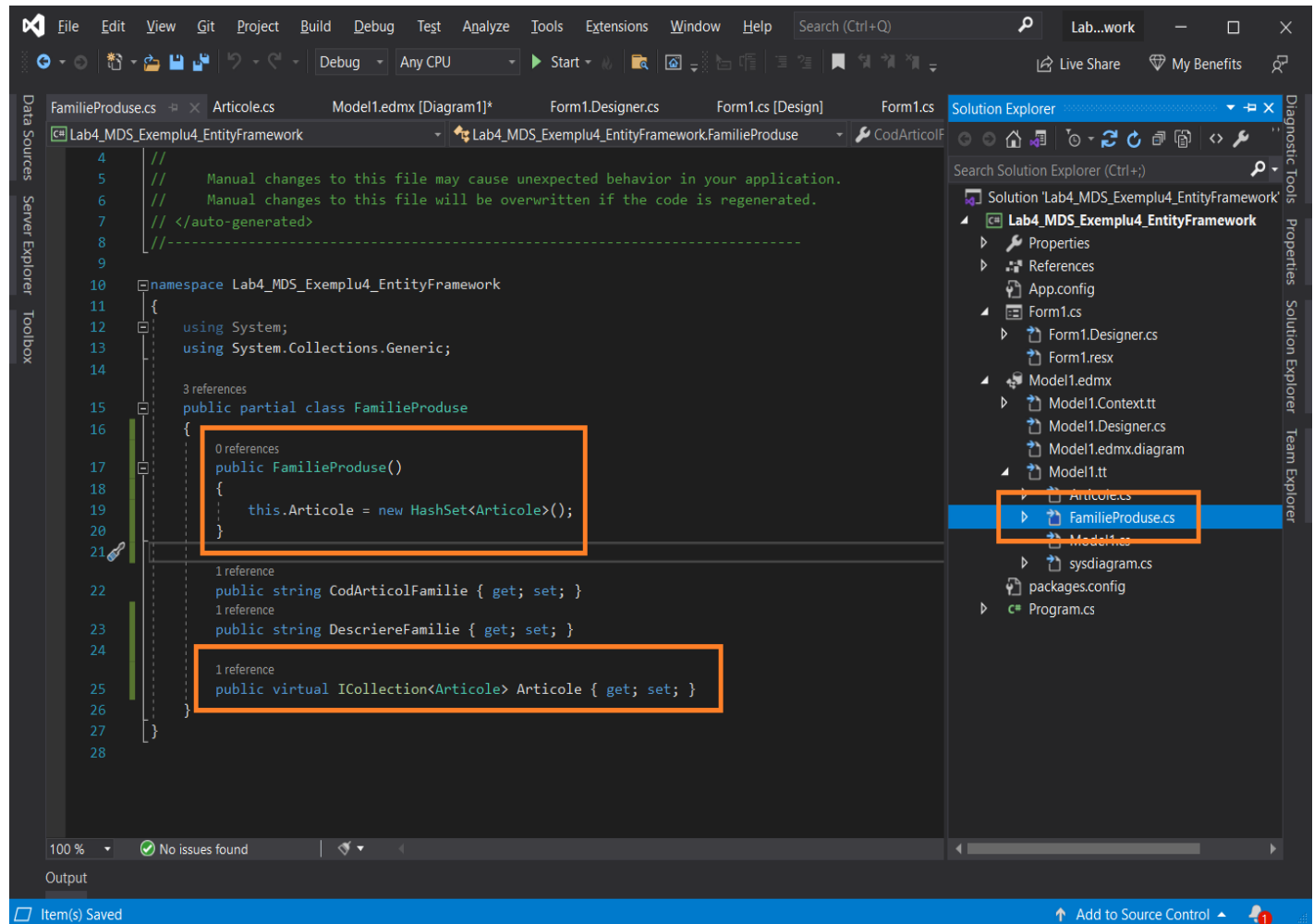


Figura 38. Modificările ce trebuie realizate

12.9. Mergeți în formularul cu cele două butoane și adăugați un al treilea buton, intitulat *Afișare Produs Foreign Key*, având următoarele proprietăți *Text* va avea valoarea *Afișare Produs Foreign Key* și (*Name*) va avea valoarea *btnAfișareProdusForeignKey*. A se vedea Figura 39 pentru detalii.

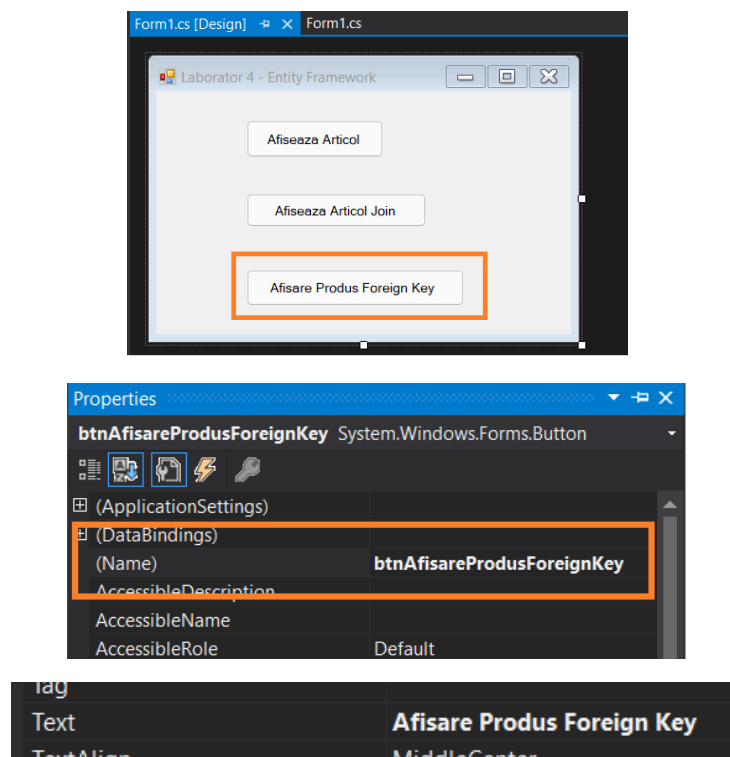


Figura 39. Butonul nou adăugat și proprietățile caracteristice

- 12.10. Adăugați codul din Figura 40. Atenție: Codul de mai jos reprezintă varianta “optimizată” pentru codul din Figura 29.

```
1reference
private void btnAfisareProdusForeignKey_Click(object sender, EventArgs e)
{
    using (MDS_Lab4Entities db = new MDS_Lab4Entities())
    {
        Articole art = db.Articole.Where((x) => x.CodArticol == "A05").FirstOrDefault();
        MessageBox.Show(art.DescriereArticol + " - " + art.FamilieProduse.DescriereFamilie);
    }
}
```

Figura 40. Codul de adăugat

- 12.11. Rulați și observați rezultatul. Comparați cu rezultatul obținut în Figura 30.

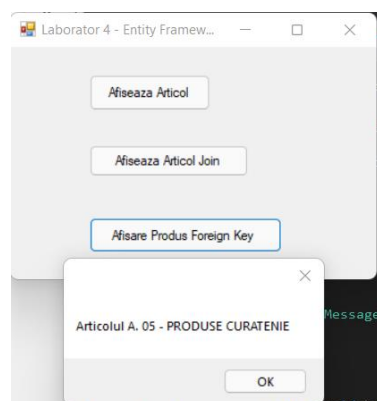


Figura 41. Rezultatul rulării