Code Academy

Dėstytojas

Vilmantas Neviera

Entity Framework core

Data

Šiandien išmoksite

01 Kas yra Entity framework core?

02 Kaip kurti duomenų bazę Code-first?

03 Kaip pridėti/atimti duomenis iš duomenų bazės?

Kas yra Entity framework’as?

https://docs.microsoft.com/en-us/ef/

Entity Framework documentation

Entity Framework Core is a modern object-database mapper for .NET. It supports LINQ queries, change tracking, updates, and schema migrations. EF Core works with many databases, including SQL Database (on-premises and Azure), SQLite, MySQL, PostgreSQL, and Azure Cosmos DB.

Entity Framework diegimas

Platesnė instrukcija kaip įsidiegti Entity framework karkasą: https://docs.microsoft.com/en-us/ef/core/get-started/overview/first-app?tabs=visual-studio

Entity Framework diegimas naudojant Visual Studio

1. Tools > NuGet Package Manager > Package Manager Console
2. Į konsolę įvedame komandą ‘Install-Package Microsoft.EntityFrameworkCore.SqlServer -Version 5.0.10’

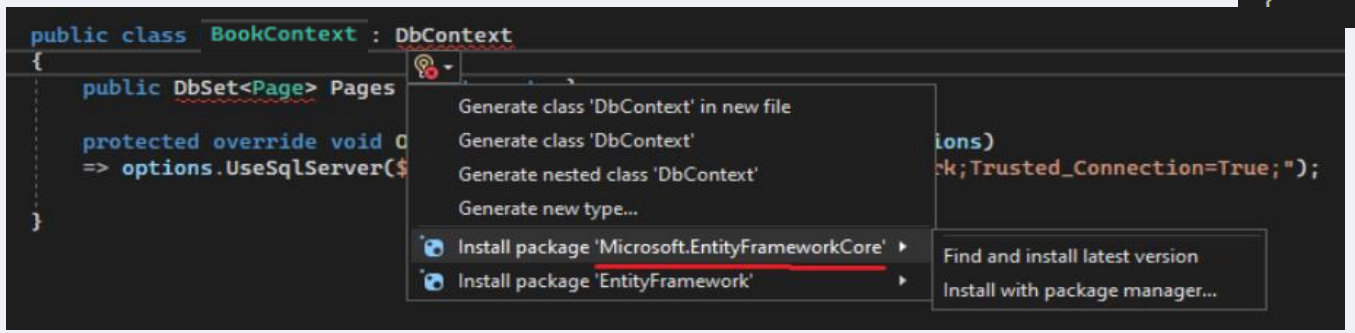
Modelio kūrimas

```
using System;

namespace EntityFrameworkPaskaita
{
    public class Page
    {
        public Guid Id { get; set; }
        public int Number { get; set; }
        public string Content { get; set; }

        public Page(int number, string content)
        {
            Id = Guid.NewGuid();
            Number = number;
            Content = content;
        }
    }
}
```

1. Susikuriame klasę pavadinimu Page
2. Susikuriame klasę pavadinimu BookContext, ši klasė bus sąsaja tarp Page klasės ir Page lentelės duomenų bazėje
3. Paveldime iš DbContext klasės ir sukuriame DbSet sąrašą Page objektų
4. Taip pat override’inam OnConfiguring metodą su savo connection string’u

```
public class BookContext : DbContext
{
    public DbSet<Page> Pages { get; set; }

    protected override void OnConfiguring(DbContextOptionsBuilder options)
    => options.UseSqlServer($"Server=localhost;Database=EntityFramework;Trusted_Connection=True;");
}
```

Atkreipkit dėmesį biblioteką pasirinkdami EntityFramework**Core**



Modelio kūrimas

Dabar galima atlikti pirmąją “Code-first” tipo migraciją.

1. Įvedame komandą ‘Install-Package Microsoft.EntityFrameworkCore.Tools -Version 5.0.10’
2. Sekanti komanda ‘Add-Migration InitialCreate’
3. Paskutinė komanda ‘Update-Database’

Modelio kūrimas

Peržvelkime kas įvyko.

Komanda Add-Migration surinko pokyčius įvykusius duomenų bazės context'e.

Šiuo atveju buvo sukurtas DbSet su Page klase.

Pati migracijos klasė atrodo taip:

Matome, kad kodas sukūrė lentelę ir stulpelius, bei PK.

Matome taip pat metodą Down, jeigu norim downgrade'int duomenų bazę, tai komanda būtų tokia:

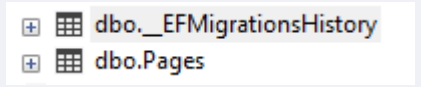
Update-Database -TargetMigration : <MigrationName>

Tokiu būdu framework'as "suktų" migracijos atgal.

```
public partial class InitialCreate : Migration
{
    protected override void Up(MigrationBuilder migrationBuilder)
    {
        migrationBuilder.CreateTable(
            name: "Pages",
            columns: table => new
            {
                Id = table.Column<Guid>(type: "uniqueidentifier", nullable: false),
                Number = table.Column<int>(type: "int", nullable: false),
                Content = table.Column<string>(type: "nvarchar(max)", nullable: true)
            },
            constraints: table =>
            {
                table.PrimaryKey("PK_Pages", x => x.Id);
            });
    }

    protected override void Down(MigrationBuilder migrationBuilder)
    {
        migrationBuilder.DropTable(
            name: "Pages");
    }
}
```

Modelio kūrimas

Atsidarę duomenų bazę, taip pat, matome sukurtą lentelę Pages ir EFMigrationsHistory



Modelio naudojimas

Pabandome pridėti duomenų

Viskas ko tam reikia tai sukurti DbContext objektą ir pridėti į DbSet sąrašą objektų.

Nepamirškit išsaugoti metodu SaveChanges()!

```
static void Main(string[] args)
{
    using var dbContext = new BookContext();

    var page = new Page(1, "text text");
    dbContext.Pages.Add(page);
    dbContext.SaveChanges();
}
```

```
SELECT TOP (1000) [Id]
      ,[Number]
      ,[Content]
  FROM [EntityFramework].[dbo].[Pages]
```

100 %

Results | Messages

	Id	Number	Content
1	19924FAA-ED20-4F70-BED0-66D7F6E4B790	1	text text

Duomenų ištrynimas

Viskas ko reikia tai sukurti objektą su Id objekto, kurį norime ištrinti ir iškviesti Remove metodą.

```
using var dbContext = new BookContext();

var page = new Page(new Guid("19924FAA-ED20-4F70-BED0-66D7F6E4B790"));
dbContext.Pages.Remove(page);
dbContext.SaveChanges();
```

Išsaugom pakeitimus context’e

Pridedame papildomą modelį

Sukurkime tėvinį modelį Page klasėi pavadinimu Book.

```
public class Book
{
    public Guid Id { get; set; }
    public string Name { get; set; }
    public List<Page> Pages { get; set; }

    public Book(string name)
    {
        Id = Guid.NewGuid();
        Name = name;
        Pages = new List<Page>();
    }
}
```

Pridėkime modelį į BookContext klasę.

```
public class BookContext : DbContext
{
    public DbSet<Page> Pages { get; set; }
    public DbSet<Book> Books { get; set; }

    protected override void OnConfiguring(DbContextOptionsBuilder options)
    => options.UseSqlServer($"Server=localhost;Database=EntityFramework;Trusted_Connection=True;");

}
```

Pridedame papildomą modelį

Atlikime naują migraciją sukelti pakeitimus.

1. Add-Migration AddedBookModel
2. Update-Database

Pridedame papildomą modelį

Dabar turime lentelę Books duomenų bazėje,

bet ji niekaip nesurišta su Page lentele.

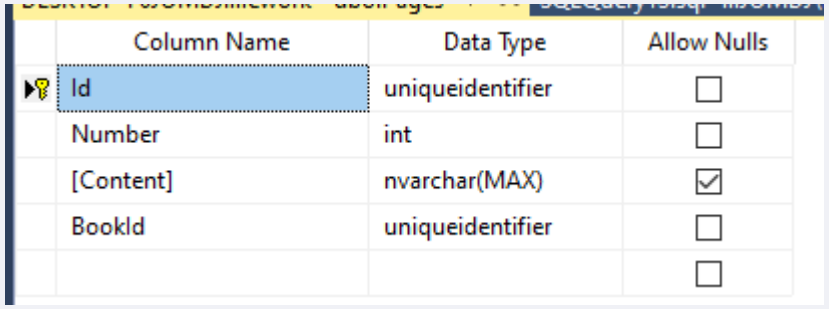
Atnaujinkime Page klasę, parodydami, kad ji yra vaikinė klasė Book klasėi.

Pridėję šiuos naujus laukus atliekame vėl naują migraciją ir duomenų bazės updatinimą.

```
public class Page
{
    public Guid Id { get; set; }
    public int Number { get; set; }
    public string Content { get; set; }

    [ForeignKey("Book")]
    public Guid BookId { get; set; }
    public Book Book { get; set; }

    public Page(int number, string content)
    {
        Id = Guid.NewGuid();
        Number = number;
        Content = content;
    }

    public Page(Guid id)
    {
        Id = id;
    }
}
```

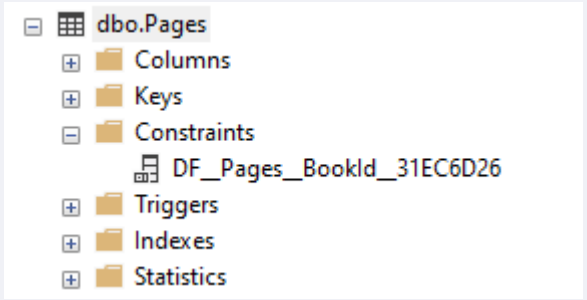
Pridedame papildomą modelį

Matome, naują lauką lentelėje ‘BookId’



	Column Name	Data Type	Allow Nulls
▶🔑	Id	uniqueidentifier	☐
	Number	int	☐
	[Content]	nvarchar(MAX)	☑
	BookId	uniqueidentifier	☐
			☐

Taip pat matome naują Constraint



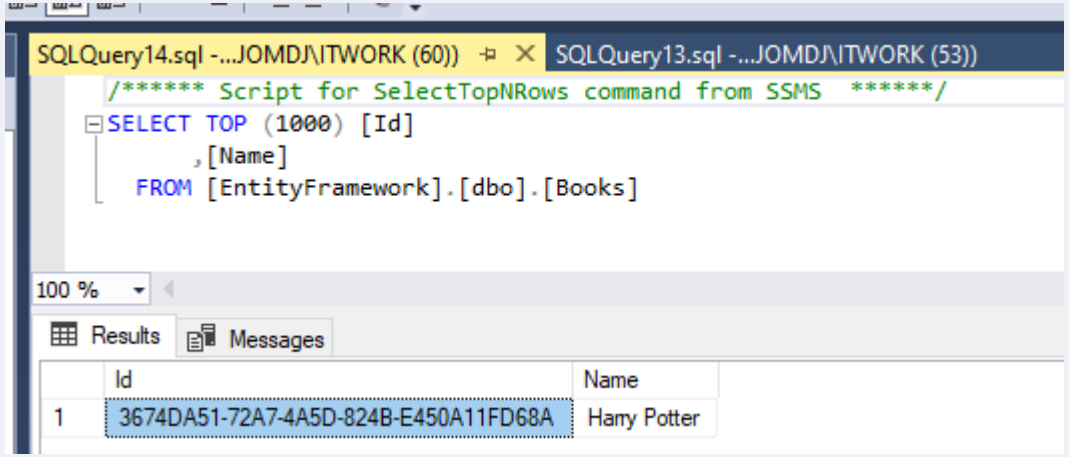
Pridedame papildomą modelį

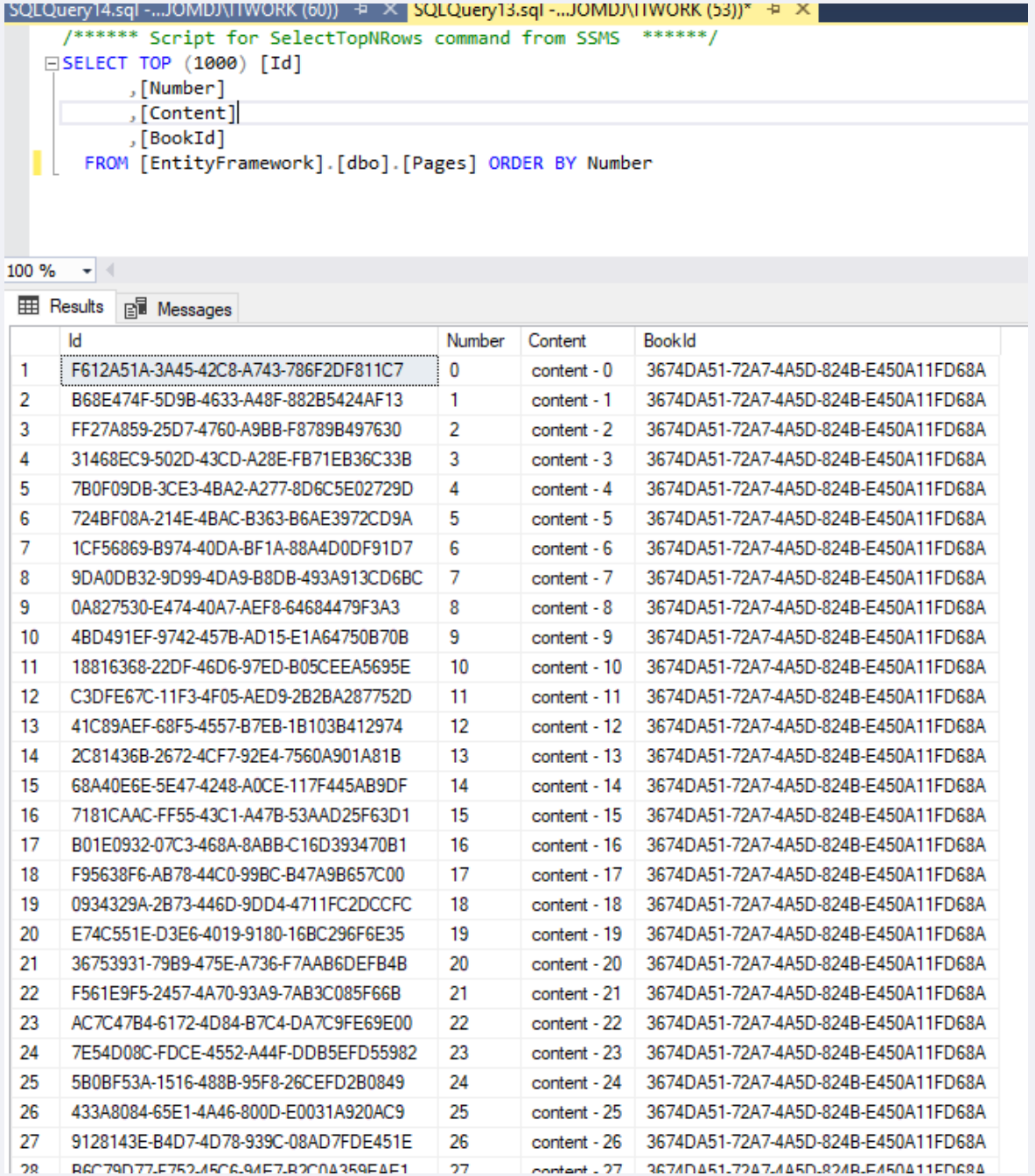
Sukuriame knygą su sąrašu puslapių ir išsaugome į duomenų bazę.

```
using var db = new PagesContext();
var book = new Book("Harry Potter");
for (int i = 0; i < 565; i++)
{
    book.Pages.Add(new Page(i, $"content - {i}"));
}

db.Books.Add(book);
db.SaveChanges();
```

Pridedame papildomą modelį

Patikriname įrašus duomenų bazėje.





	Id	Number	Content	BookId
1	F612A51A-3A45-42C8-A743-786F2DF811C7	0	content - 0	3674DA51-72A7-4A5D-824B-E450A11FD68A
2	B68E474F-5D9B-4633-A48F-882B5424AF13	1	content - 1	3674DA51-72A7-4A5D-824B-E450A11FD68A
3	FF27A859-25D7-4760-A9BB-F8789B497630	2	content - 2	3674DA51-72A7-4A5D-824B-E450A11FD68A
4	31468EC9-502D-43CD-A28E-FB71EB36C33B	3	content - 3	3674DA51-72A7-4A5D-824B-E450A11FD68A
5	7B0F09DB-3CE3-4BA2-A277-8D6C5E02729D	4	content - 4	3674DA51-72A7-4A5D-824B-E450A11FD68A
6	724BF08A-214E-4BAC-B363-B6AE3972CD9A	5	content - 5	3674DA51-72A7-4A5D-824B-E450A11FD68A
7	1CF56869-B974-40DA-BF1A-88A4D0DF91D7	6	content - 6	3674DA51-72A7-4A5D-824B-E450A11FD68A
8	9DA0DB32-9D99-4DA9-B8DB-493A913CD6BC	7	content - 7	3674DA51-72A7-4A5D-824B-E450A11FD68A
9	0A827530-E474-40A7-AEF8-64684479F3A3	8	content - 8	3674DA51-72A7-4A5D-824B-E450A11FD68A
10	4BD491EF-9742-457B-AD15-E1A64750B70B	9	content - 9	3674DA51-72A7-4A5D-824B-E450A11FD68A
11	18816368-22DF-46D6-97ED-B05CEEA5695E	10	content - 10	3674DA51-72A7-4A5D-824B-E450A11FD68A
12	C3DFE67C-11F3-4F05-AED9-2B2BA287752D	11	content - 11	3674DA51-72A7-4A5D-824B-E450A11FD68A
13	41C89AEF-68F5-4557-B7EB-1B103B412974	12	content - 12	3674DA51-72A7-4A5D-824B-E450A11FD68A
14	2C81436B-2672-4CF7-92E4-7560A901A81B	13	content - 13	3674DA51-72A7-4A5D-824B-E450A11FD68A
15	68A40E6E-5E47-4248-A0CE-117F445AB9DF	14	content - 14	3674DA51-72A7-4A5D-824B-E450A11FD68A
16	7181CAAC-FF55-43C1-A47B-53AAD25F63D1	15	content - 15	3674DA51-72A7-4A5D-824B-E450A11FD68A
17	B01E0932-07C3-468A-8ABB-C16D393470B1	16	content - 16	3674DA51-72A7-4A5D-824B-E450A11FD68A
18	F95638F6-AB78-44C0-99BC-B47A9B657C00	17	content - 17	3674DA51-72A7-4A5D-824B-E450A11FD68A
19	0934329A-2B73-446D-9DD4-4711FC2DCCFC	18	content - 18	3674DA51-72A7-4A5D-824B-E450A11FD68A
20	E74C551E-D3E6-4019-9180-16BC296F6E35	19	content - 19	3674DA51-72A7-4A5D-824B-E450A11FD68A
21	36753931-79B9-475E-A736-F7AAB6DEFB4B	20	content - 20	3674DA51-72A7-4A5D-824B-E450A11FD68A
22	F561E9F5-2457-4A70-93A9-7AB3C085F66B	21	content - 21	3674DA51-72A7-4A5D-824B-E450A11FD68A
23	AC7C47B4-6172-4D84-B7C4-DA7C9FE69E00	22	content - 22	3674DA51-72A7-4A5D-824B-E450A11FD68A
24	7E54D08C-FDCE-4552-A44F-DDB5EFD55982	23	content - 23	3674DA51-72A7-4A5D-824B-E450A11FD68A
25	5B0BF53A-1516-488B-95F8-26CEFD2B0849	24	content - 24	3674DA51-72A7-4A5D-824B-E450A11FD68A
26	433A8084-65E1-4A46-800D-E0031A920AC9	25	content - 25	3674DA51-72A7-4A5D-824B-E450A11FD68A
27	9128143E-B4D7-4D78-939C-08AD7FDE451E	26	content - 26	3674DA51-72A7-4A5D-824B-E450A11FD68A
28	B6C79D77-F752-45C6-94E7-B2C0A359EAE1	27	content - 27	3674DA51-72A7-4A5D-824B-E450A11FD68A

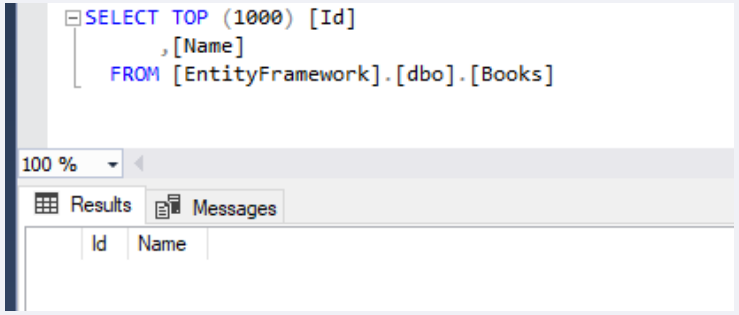
Ištrynimas tėvų(modelių) su vaikais.

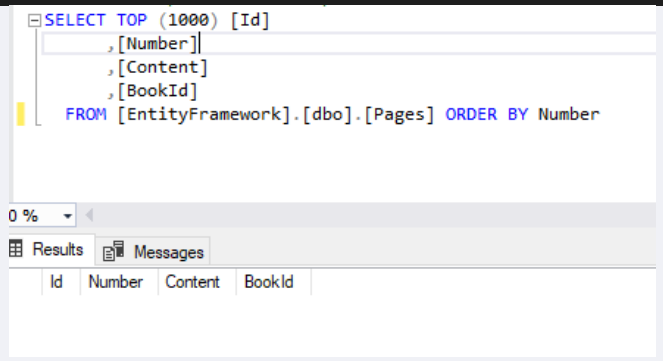
Norėdami ištrinti tėvinį elementą su vaikais, turime pradžioje juos išsitraukti kartu iš duomenų bazės.

Tam reikės naudoti LINQ Where ir Include metodus.

```
static void Main(string[] args)
{
    using var db = new PagesContext();
    var book = db.Books.Where(x => x.Id == new Guid("3674DA51-72A7-4A5D-824B-E450A11FD68A")).Include(x => x.Pages).First();

    db.Books.Remove(book);
    db.SaveChanges();

}
```

Kaip matome duomenys buvo ištrinti:





Užduotis nr. 1

- Perdarome 1lvl pradmenų patikrinimo užduotį, kad vietoj failų naudotų duomenų bazę;

Headline

www.youtube.com

Naudinga informacija