**Задание 10**

Создать консольное приложение, с помощью которого осуществить

подключение к базе данных, используя фреймворк EF Core

(использовать подход CodeFirst). Создать две связанные сущности,

реализуя связь один-много. Наполнить БД значениями и вывести на

консоль список всех элементов в БД с загрузкой связанных сущностей

используя подход Lazy Loading (ленивая загрузка)

**Код программы:**

1. Добавить пакеты:

Microsoft.EntityFrameworkCore

Microsoft.EntityFrameworkCore.SqlServer

2. Создаем классы Author, Book, LibraryContext.

3. Код Author:

public class Author

{

public int Id { get; set; }

public string Name { get; set; }

public virtual ICollection<Book> Books { get; set; }

}

4. Код Book:

public class Book

{

public int Id { get; set; }

public string Title { get; set; }

public int AuthorId { get; set; }

public virtual Author Author { get; set; }

}

5. Код LibraryContext:

public class LibraryContext : DbContext

{

public DbSet<Author> Authors { get; set; }

public DbSet<Book> Books { get; set; }

protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)

{

optionsBuilder.UseSqlServer("Server=(localdb)\\mssqllocaldb;Database=Library;Trusted\_Connection=True;");

}

protected override void OnModelCreating(ModelBuilder modelBuilder)

{

modelBuilder.Entity<Book>()

.HasOne(b => b.Author)

.WithMany(a => a.Books)

.HasForeignKey(b => b.AuthorId);

}

}

6. Код Programm:

using Microsoft.EntityFrameworkCore;

class Program

{

static void Main(string[] args)

{

// Создаем и заполняем базу данных

using (var db = new LibraryContext())

{

db.Database.EnsureDeleted();

db.Database.EnsureCreated();

var author1 = new Author { Name = "Leo Tolstoy" };

var author2 = new Author { Name = "Fyodor Dostoevsky" };

db.Authors.AddRange(author1, author2);

var book1 = new Book { Title = "War and Peace", Author = author1 };

var book2 = new Book { Title = "Anna Karenina", Author = author1 };

var book3 = new Book { Title = "Crime and Punishment", Author = author2 };

var book4 = new Book { Title = "The Brothers Karamazov", Author = author2 };

db.Books.AddRange(book1, book2, book3, book4);

db.SaveChanges();

}

// Выводим список всех книг с указанием автора

using (var db = new LibraryContext())

{

var books = db.Books.Include(b => b.Author).ToList();

foreach (var book in books)

{

string authorName = book.Author != null ? book.Author.Name : "Unknown";

Console.WriteLine($"Title: {book.Title}, Author: {authorName}");

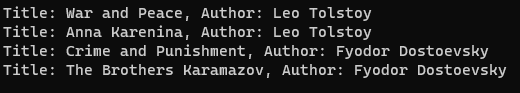
}

}

}

}

**Результат работы программы:**

****

