**33 Технология LINQ to SQL**

Задание №1. Выполните операции просмотра, изменения, добавления и удаления данных с помощью технологии LinqToSQL. Принцип работы описан в документе «Пример LINQ to SQL.docx»

Листинг программы:

using Microsoft.EntityFrameworkCore;

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations.Schema;

namespace Task

{

public class YourDbContext : DbContext

{

public DbSet<Tour> Tours { get; set; }

public DbSet<Tourist> Tourists { get; set; }

protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)

{

optionsBuilder.UseSqlite("Data Source=D:\\!!!!!!!ПРАКТИКА\\Виноградова\_день34\\Практическая часть34\\DBTur\_firm.sqlite");

}

}

[Table("Туры")]

public class Tour

{

[Column("Код\_тура")]

public int TourId { get; set; }

[Column("Название")]

public string Name { get; set; }

[Column("Дата\_начала")]

public DateTime StartDate { get; set; }

[Column("Дата\_окончания")]

public DateTime EndDate { get; set; }

[Column("Цена")]

public decimal Price { get; set; }

}

[Table("Туристы")]

public class Tourist

{

[Column("Код\_туриста")]

public int TouristId { get; set; }

[Column("Фамилия")]

public string LastName { get; set; }

[Column("Имя")]

public string FirstName { get; set; }

[Column("Отчество")]

public string MiddleName { get; set; }

[Column("Код\_тура")]

public int TourId { get; set; }

}

class Program

{

static void Main()

{

using (var db = new YourDbContext())

{

var tourists = db.Tourists.ToList();

Console.WriteLine("Список туристов:");

foreach (var tourist in tourists)

{

Console.WriteLine($"ID: {tourist.TouristId}, Фамилия: {tourist.LastName}, Имя: {tourist.FirstName}, Отчество: {tourist.MiddleName}, ID тура: {tourist.TourId}");

}

}

}

}

}

Анализ результатов:

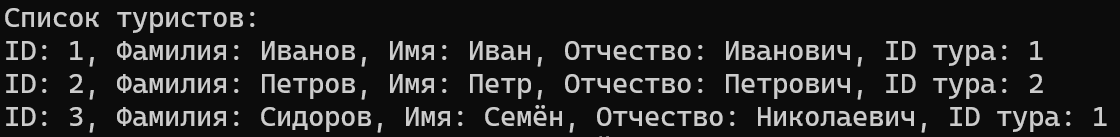


Рисунок 33.1 – Результаты работы программы