**33 ТЕХНОЛОГИЯ LINQ TO SQL**

Задание 1. Выполните операции просмотра, добавления и удаления данных с помощью технологии LinqToSQL.

Листинг программы

public class Payment

{ [DatabaseGenerated(DatabaseGeneratedOption.Identity)]

public int PaymentId { get; set; }

public int TicketId { get; set; }

public DateTime Date { get; set; }

public decimal Sum { get; set; }

public virtual Ticket Ticket { get; set; }

}

public class Season

{ [DatabaseGenerated(DatabaseGeneratedOption.Identity)]

public int SeasonId { get; set; }

public int TicketId { get; set; }

public DateTime Start { get; set; }

public DateTime End { get; set; }

public bool IsClosed { get; set; }

public int Count { get; set; }

public virtual Ticket Ticket { get; set; }

public virtual ICollection<Tur> Turs { get; set; } = new HashSet<Tur>();

}

public class Ticket

{ [DatabaseGenerated(DatabaseGeneratedOption.Identity)]

public int TicketId { get; set; }

public virtual ICollection<Season> Seasons { get; set; } = new HashSet<Season>();

public virtual ICollection<TuristInformation> TuristInformations { get; set; } = new HashSet<TuristInformation>();

}

public class Tourist

{ [DatabaseGenerated(DatabaseGeneratedOption.Identity)]

public int TouristId { get; set; } public string FirstName { get; set; }

public string SecondName { get; set; }

public string LastName { get; set; }

}

public class Tur

{ [DatabaseGenerated(DatabaseGeneratedOption.Identity)]

public int TurId { get; set; }

public int SeasonId { get; set; }

public string Name { get; set; }

public decimal Cost { get; set; }

public string Information { get; set; }

public virtual Season Season { get; set; }

}

public class TuristInformation

{ [DatabaseGenerated(DatabaseGeneratedOption.Identity)]

public int TuristInformationId { get; set; }

public int TirustId { get; set; }

public string SerialNumber { get; set; }

public string City { get; set; }

public string Country { get; set; }

public string PhoneNumber { get; set; }

public string Index { get; set; }

public virtual Tourist Turists { get; set; }

}

internal class AppDbContext : DbContext

{public DbSet<Payment> Payments { get; set; }

public DbSet<Season> Seasons { get; set; }

public DbSet<Ticket> Tickets { get; set; }

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

public DbSet<Tur> Turs { get; set; }

public DbSet<TuristInformation> TuristInformations { get; set; }

public AppDbContext(DbContextOptions<AppDbContext> options): base(options)

{

}

public MainWindow()

{InitializeComponent();

\_context = new AppDbContext(new DbContextOptionsBuilder<AppDbContext>()

.UseSqlServer("Data Source=DESKTOP-T9IBF8L;Initial Catalog=lab34\_1;Integrated Security=SSPI;")

.Options);

\_context.Database.EnsureCreated();

RefreshTuristsGrid();

}

private void Button\_Click(object sender, RoutedEventArgs e)

{var dialog = new TuristWindow();

dialog.ShowDialog();

if (dialog.DialogResult == true)

{ \_context.Turists.Add(dialog.Tourist);

\_context.SaveChanges();

}

RefreshTuristsGrid();

}

private void RefreshTuristsGrid()

{TouristGrid.DataContext = \_context.Turists.ToList();

}

private void Deletebutton\_Click(object sender, RoutedEventArgs e)

{var item = TouristGrid.SelectedItem as Tourist;

var entity = \_context.Turists.First(x => x.TouristId == item.TouristId);

\_context.Turists.Remove(entity);

\_context.SaveChanges();

RefreshTuristsGrid();

}

Анализ результатов:

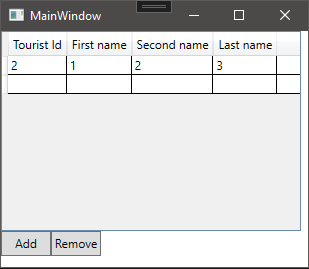


Рисунок 33.1 – Результат работы программы

Источник: собственная разработка