МДК 01.03 «Разработка мобильных приложений»

Практическая работа №4

Учебная группа 32.1Исд

Студент Данилин Николай Юрьевич

# Код программного приложения

Листинг 1.

|  |
| --- |
| MainViewModel  using danilin\_praktika.Contexts;  using danilin\_praktika.Models;  using System;  using System.Collections.Generic;  using System.ComponentModel;  using System.Linq;  using System.Runtime.CompilerServices;  using System.Text;  using System.Threading.Tasks;  namespace danilin\_praktika.ViewModels  {  public class MainViewModel : INotifyPropertyChanged  {  public event PropertyChangedEventHandler PropertyChanged;  public void OnPropertyChanged([CallerMemberName] string propertyName = "")  {  PropertyChanged?.Invoke(this, new PropertyChangedEventArgs(propertyName));  }  private IEnumerable<Author> \_authors;  public IEnumerable<Author> AUTRORS  {  get  {  return \_authors;  }  set  {  \_authors = value;  OnPropertyChanged();  }  }  private void GetAuthor()  {  using (var context = new AnyContext())  {  \_authors = context.Authors.ToList();  }  }  private IEnumerable<Book> \_books;  public IEnumerable<Book> BOOKS  {  get  {  return \_books;  }  set  {  \_books = value;  OnPropertyChanged();  }  }  private Book \_newBook;  public Book newBook  {  get  {  return \_newBook;  }  set  {  \_newBook = value;  OnPropertyChanged();  }  }  private Author \_newAuthor;  public Author newAuthor  {  get  {  return \_newAuthor;  }  set  {  \_newAuthor = value;  OnPropertyChanged();  }  }  private void GetBook()  {  using (var context = new AnyContext())  {  \_books = context.Books.ToList();  }  }  public DelegateCommand AddNewAuthorCommand  {  get  {  return new DelegateCommand(o =>  {  AddNewAuthor();  });  }  }  private void AddNewAuthor()  {  using (var context = new AnyContext())  {  context.Authors.Add(newAuthor);  context.SaveChanges();  AUTRORS = context.Authors.ToList();  }  newAuthor = new Author();  }  public DelegateCommand AddNewBookCommand  {  get  {  return new DelegateCommand(o =>  {  AddNewBook();  });  }  }  private void AddNewBook()  {  using (var context = new AnyContext())  {  context.Books.Add(newBook);  context.SaveChanges();  BOOKS = context.Books.ToList();  }  newBook = new Book();  }  private void DeleteBook(int bookId)  {  using (var context = new AnyContext())  {  Book BookForDelete = context.Books.Find(bookId);  context.Books.Remove(BookForDelete);  context.SaveChanges();  BOOKS = context.Books.ToList();  }  }  public DelegateCommand DeleteBookCommand  {  get  {  return new DelegateCommand(o =>  {  DeleteBook((int)o);  });  }  }  public MainViewModel()  {  newAuthor = new Author();  newBook = new Book();  GetAuthor();  GetBook();  }  }  } |
| AnyContext |
| using danilin\_praktika.Models;  using Microsoft.EntityFrameworkCore;  using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace danilin\_praktika.Contexts  {  public class AnyContext : DbContext  {  public DbSet<Author> Authors { get; set; }  public DbSet<Book> Books { get; set; }  protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)  {  optionsBuilder.UseSqlServer("Server=Nikolay\\SQLEXPRESS;Database=danilin\_MDK0104\_5;Trusted\_Connection=True;TrustServerCertificate=True");  }  }  }  AUTHOR |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace danilin\_praktika.Models  {  public class Author  {  public int authorId { get; set; }  public string authorName { get; set; }  public DateTime authorBorn { get; set; }  public int publishedBooks { get; set; }  public string authorCountry { get; set; }  }  }  BOOK  using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace danilin\_praktika.Models  {  public class Book  {  public int bookId { get; set; }  public string bookName { get; set; }  public int bookAuthor { get; set; }  public int bookPages { get; set; }  public string bookGenre { get; set; }  }  }  XAML  <Window x:Class="danilin\_praktika.MainWindow"  xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"  xmlns:d="http://schemas.microsoft.com/expression/blend/2008"  xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"  xmlns:local="clr-namespace:danilin\_praktika"  xmlns:vm="clr-namespace:danilin\_praktika.ViewModels"  mc:Ignorable="d"  Title="MainWindow" Height="450" Width="800" Background="#FFF7E696">  <Window.DataContext>  <vm:MainViewModel/>  </Window.DataContext>  <StackPanel Orientation="Horizontal" HorizontalAlignment="Center" Background="White">  <StackPanel>  <ListView ItemsSource="{Binding AUTRORS}">  <ListView.View>  <GridView>  <GridViewColumn Header="authorId" DisplayMemberBinding="{Binding authorId}"></GridViewColumn>  <GridViewColumn Header="authorName" DisplayMemberBinding="{Binding authorName}"></GridViewColumn>  <GridViewColumn Header="authorAge" DisplayMemberBinding="{Binding authorBorn}"></GridViewColumn>  <GridViewColumn Header="authorCountry" DisplayMemberBinding="{Binding authorCountry}"></GridViewColumn>  </GridView>  </ListView.View>  </ListView>  <GroupBox>  <StackPanel>  <StackPanel>  <Label Content="Имя автора"/>  <TextBox Text="{Binding newAuthor.authorName}"/>  </StackPanel>  <StackPanel>  <Label Content="Возраст автора"/>  <TextBox Text="{Binding newAuthor.authorBorn}"/>  </StackPanel>  <StackPanel>  <Label Content="Страна автора"/>  <TextBox Text="{Binding newAuthor.authorCountry}"/>  </StackPanel>  <Button Command="{Binding AddNewAuthorCommand}" Content="Добавить автора" Background="#FFA0E274" BorderBrush="#FFB6FF7C" Foreground="White" FontWeight="Bold"/>  </StackPanel>  </GroupBox>  </StackPanel>  <StackPanel>  <ListView x:Name="listView" ItemsSource="{Binding BOOKS}">  <ListView.View>  <GridView>  <GridViewColumn Header="bookId" DisplayMemberBinding="{Binding bookId}"></GridViewColumn>  <GridViewColumn Header="bookName" DisplayMemberBinding="{Binding bookName}"></GridViewColumn>  <GridViewColumn Header="bookGenre" DisplayMemberBinding="{Binding bookGenre}"></GridViewColumn>  <GridViewColumn Header="bookPages" DisplayMemberBinding="{Binding bookPages}"></GridViewColumn>  <GridViewColumn Header="authorId" DisplayMemberBinding="{Binding bookAuthor}"></GridViewColumn>  <GridViewColumn>  <GridViewColumn.CellTemplate>  <DataTemplate>  <Button Width="50" Height="20" Command="{Binding ElementName=listView, Path=DataContext.DeleteBookCommand}" CommandParameter="{Binding bookId}" Content="Удалить" FontSize="12" FontWeight="Bold" BorderBrush="#FFFF7979" Background="#FFFF7D7D" Foreground="White"></Button>  </DataTemplate>  </GridViewColumn.CellTemplate>  </GridViewColumn>  </GridView>  </ListView.View>  </ListView>  <GroupBox>  <StackPanel>  <StackPanel>  <Label Content="Название книги"/>  <TextBox Text="{Binding newBook.bookName}"/>  </StackPanel>  <StackPanel>  <Label Content="Жанр книги"/>  <TextBox Text="{Binding newBook.bookGenre}"/>  </StackPanel>  <StackPanel>  <Label Content="Кол-во страниц"/>  <TextBox Text="{Binding newBook.bookPages}"/>  </StackPanel>  <StackPanel>  <Label Content="Номер автора"/>  <TextBox Text="{Binding newBook.bookAuthor}"/>  </StackPanel>  <Button Command="{Binding AddNewBookCommand}" Content="Добавить книгу"/>  </StackPanel>  </GroupBox>  </StackPanel>  </StackPanel>  </Window> |

# Демонстрация работы приложения

Видео находится в гитхабе под названием «ВИДЕО Практическая 4»