Main Page

<StackLayout Padding="20">

<Label Text="Введите данные о студенте" FontSize="20" HorizontalOptions="CenterAndExpand" Margin="0,0,0,20" />

<Entry Placeholder="Фамилия" x:Name="LastNameEntry" />

<Entry Placeholder="Имя" x:Name="FirstNameEntry" />

<Entry Placeholder="Группа" x:Name="GroupEntry" />

<Entry Placeholder="Курс" x:Name="CourseEntry" Keyboard="Numeric" />

<Picker x:Name="GenderPicker" Title="Пол">

<Picker.ItemsSource>

<x:Array Type="{x:Type x:String}">

<x:String>Мужской</x:String>

<x:String>Женский</x:String>

</x:Array>

</Picker.ItemsSource>

</Picker>

<Entry Placeholder="Дата рождения" x:Name="BirthEntry" />

<Button Text="Сохранить" Clicked="SaveButton\_Clicked" HorizontalOptions="Center" Margin="0,20,0,0" />

</StackLayout>

</ContentPage>

namespace Bil1

{

public partial class MainPage : ContentPage

{

private string filePath;

public MainPage()

{

InitializeComponent();

filePath = Path.Combine(Environment.GetFolderPath(Environment.SpecialFolder.LocalApplicationData), "students.txt");

}

private async void SaveButton\_Clicked(object sender, EventArgs e)

{

string lastName = LastNameEntry.Text;

string firstName = FirstNameEntry.Text;

string group = GroupEntry.Text;

int course = int.Parse(CourseEntry.Text);

string gender = GenderPicker.SelectedItem.ToString();

string birthDate = BirthEntry.Text;

// Проверка корректности данных

if (string.IsNullOrWhiteSpace(lastName) || string.IsNullOrWhiteSpace(firstName) || string.IsNullOrWhiteSpace(group))

{

await DisplayAlert("Ошибка", "Пожалуйста, заполните все поля.", "OK");

return;

}

// Сохранение данных в файл

string studentData = $"{lastName},{firstName},{group},{course},{gender},{birthDate}";

File.AppendAllText(filePath, studentData + Environment.NewLine);

await DisplayAlert("Успех", "Данные студента сохранены.", "OK");

await Navigation.PushAsync(new StudentsPage());

// Очистка полей ввода

LastNameEntry.Text = string.Empty;

FirstNameEntry.Text = string.Empty;

GroupEntry.Text = string.Empty;

CourseEntry.Text = string.Empty;

GenderPicker.SelectedItem = null;

BirthEntry.Text = string.Empty;

}

}

}

StudentsPage

<StackLayout Padding="20">

<Label Text="Сохраненные данные о студентах" FontSize="20" HorizontalOptions="CenterAndExpand" Margin="0,0,0,20" />

<ListView x:Name="StudentsListView" RowHeight="150">

<ListView.ItemTemplate>

<DataTemplate>

<ViewCell>

<StackLayout Padding="10" Orientation="Vertical">

<Label Text="{Binding LastName}" FontAttributes="Bold" />

<Label Text="{Binding FirstName}" />

<Label Text="{Binding Group}" />

<Label Text="{Binding Course}" />

<Label Text="{Binding Gender}" />

<Label Text="{Binding BirthDate}" />

</StackLayout>

</ViewCell>

</DataTemplate>

</ListView.ItemTemplate>

</ListView>

</StackLayout>

</ContentPage>

namespace Bil1

{

public partial class StudentsPage : ContentPage

{

private string filePath;

public StudentsPage()

{

InitializeComponent();

filePath = Path.Combine(Environment.GetFolderPath(Environment.SpecialFolder.LocalApplicationData), "students.txt");

}

protected override void OnAppearing()

{

base.OnAppearing();

// Чтение данных из файла

string[] studentData = File.ReadAllLines(filePath);

// Создание списка студентов

var students = studentData.Select(data =>

{

string[] fields = data.Split(',');

return new

{

LastName = fields[0],

FirstName = fields[1],

Group = fields[2],

Course = int.Parse(fields[3]),

Gender = fields[4],

BirthDate = fields[5]

};

}).ToList();

// Отображение списка студентов в ListView

StudentsListView.ItemsSource = students;

}

}

}

App xaml cs

MainPage = new NavigationPage(new MainPage());