Министерство образования и молодежной политики Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчет по ТРПО. Лабораторная работа   
**Анимация в WPF-приложениях**

Выполнили: Максимов К. Э.

Группа: ПР-31

Преподаватель: Бушуева Е. В.

2024

Задание 1

<Window x:Class="Lab\_5\_WPF.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:Lab\_5\_WPF"

mc:Ignorable="d"

Title="MultiEdit" Height="450" Width="800" Loaded="Window\_Loaded">

<Grid>

<Grid.ColumnDefinitions>

<ColumnDefinition Width="\*" />

<ColumnDefinition Width="\*" />

</Grid.ColumnDefinitions>

<StackPanel x:Name="LeftPanel" Grid.Column="0" Margin="10"/>

<StackPanel x:Name="RightPanel" Grid.Column="1" Margin="10"/>

</Grid>

</Window>

namespace Lab\_5\_WPF

{

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

private void Window\_Loaded(object sender, RoutedEventArgs e)

{

// Создание градиента

LinearGradientBrush gradientBrush = new LinearGradientBrush();

gradientBrush.GradientStops.Add(new GradientStop(Colors.LightBlue, 0));

gradientBrush.GradientStops.Add(new GradientStop(Colors.White, 1));

// Применение градиента к двум панелям

LeftPanel.Background = gradientBrush;

RightPanel.Background = gradientBrush;

// Установка стилей для текстовых полей

Style largeTextBoxStyle = new Style(typeof(TextBox));

largeTextBoxStyle.Setters.Add(new Setter(TextBox.FontSizeProperty, 18.0));

Style smallTextBoxStyle = new Style(typeof(TextBox));

smallTextBoxStyle.Setters.Add(new Setter(TextBox.FontSizeProperty, 12.0));

// Создание текстовых полей

TextBox largeTextBox1 = new TextBox();

largeTextBox1.Style = largeTextBoxStyle;

TextBox largeTextBox2 = new TextBox();

largeTextBox2.Style = largeTextBoxStyle;

TextBox smallTextBox1 = new TextBox();

smallTextBox1.Style = smallTextBoxStyle;

TextBox smallTextBox2 = new TextBox();

smallTextBox2.Style = smallTextBoxStyle;

// Добавление текстовых полей в панели

LeftPanel.Children.Add(largeTextBox1);

LeftPanel.Children.Add(smallTextBox1);

RightPanel.Children.Add(largeTextBox2);

RightPanel.Children.Add(smallTextBox2);

// Создание анимации

DoubleAnimation animation = new DoubleAnimation();

animation.From = 0.0;

animation.To = 1.0;

animation.Duration = new Duration(TimeSpan.FromSeconds(2));

// Применение анимации к текстовым полям

largeTextBox1.BeginAnimation(TextBox.OpacityProperty, animation);

largeTextBox2.BeginAnimation(TextBox.OpacityProperty, animation);

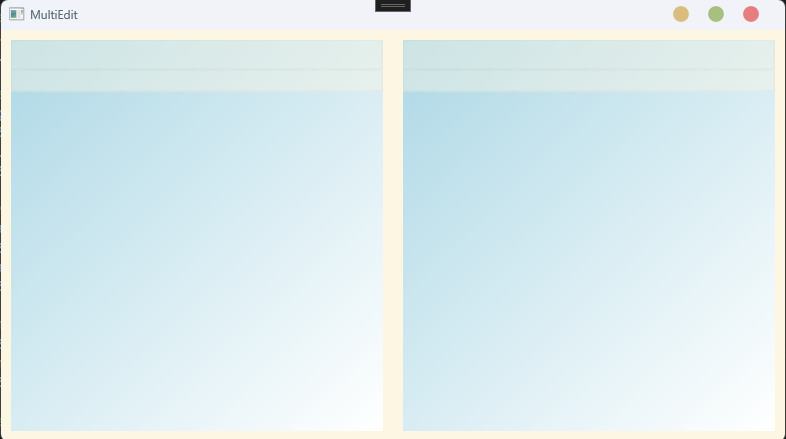
smallTextBox1.BeginAnimation(TextBox.OpacityProperty, animation);

smallTextBox2.BeginAnimation(TextBox.OpacityProperty, animation);

}

}

}



Задание 2

<Window x:Class="Lab\_7\_WPF.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:Lab\_7\_WPF"

mc:Ignorable="d"

Title="Убегающая кнопка" Height="350" Width="525">

<Grid>

<Button x:Name="MyButton" Content="Нажми меня" MouseEnter="Button\_MouseEnter" Width="100" Height="50"/>

</Grid>

</Window>

namespace Lab\_7\_WPF

{

public partial class MainWindow : Window

{

private Random \_random = new Random();

public MainWindow()

{

InitializeComponent();

}

private void Button\_MouseEnter(object sender, MouseEventArgs e)

{

double windowWidth = this.Width;

double windowHeight = this.Height;

double buttonWidth = MyButton.ActualWidth;

double buttonHeight = MyButton.ActualHeight;

double left = \_random.NextDouble() \* (windowWidth - buttonWidth);

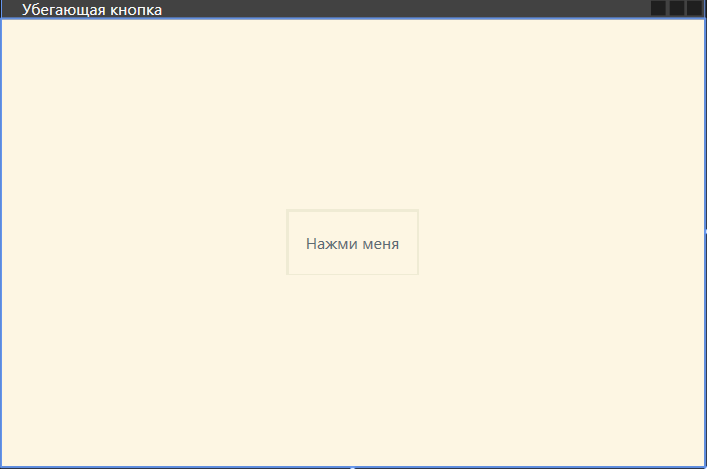
double top = \_random.NextDouble() \* (windowHeight - buttonHeight);

MyButton.Margin = new Thickness(left, top, 0, 0);

}

}

}



Задание 3

<Window x:Class="PulsarApp.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

Title="Pulsar" Width="400" Height="400">

<Grid>

<Ellipse x:Name="MyEllipse" Width="300" Height="300" HorizontalAlignment="Center" VerticalAlignment="Center"/>

</Grid>

</Window>

namespace PulsarApp

{

public partial class MainWindow : Window

{

private RadialGradientBrush \_gradientBrush;

private DoubleAnimation \_animation;

public MainWindow()

{

InitializeComponent();

\_gradientBrush = new RadialGradientBrush();

\_gradientBrush.GradientStops.Add(new GradientStop(Colors.Red, 0.0));

\_gradientBrush.GradientStops.Add(new GradientStop(Colors.Orange, 0.5));

\_gradientBrush.GradientStops.Add(new GradientStop(Colors.Yellow, 1.0));

MyEllipse.Fill = \_gradientBrush;

\_animation = new DoubleAnimation();

\_animation.From = 0.0;

\_animation.To = 1.0;

\_animation.Duration = new Duration(TimeSpan.FromSeconds(1));

\_animation.AutoReverse = true;

\_animation.RepeatBehavior = RepeatBehavior.Forever;

\_gradientBrush.GradientStops[1].BeginAnimation(GradientStop.OffsetProperty, \_animation);

}

}

}

