Частное учреждение образования

«Колледж бизнеса и права»

ОТЧЕТ ПО САМОСТОЯТЕЛЬНОЙ РАБОТЕ №3

Т.091005

Руководитель практики Е.Н.Коропа

Учащийся А.О.Зеневич

2022

**«WPF»**

**ВАРИАНТ 4**

MainWindow.xaml

<Window x:Class="Prac08.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:Prac08"

mc:Ignorable="d"

Title="MainWindow" Height="450" Width="800" Closing="Window\_Closing">

<Grid>

<Menu Height="NaN" Width="NaN">

<MenuItem Header="FirstTask" Click="FirstTask\_Click"/>

<MenuItem Header="SecondTask" Click="SecondTask\_Click"/>

</Menu>

</Grid>

</Window>

MainWindow.xaml.cs

using System.Windows;

namespace Prac08

{

/// <summary>

/// Логика взаимодействия для MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

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

{

this.Hide();

FirstTask firstTask = new FirstTask();

firstTask.Show();

}

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

{

this.Hide();

SecondTask secondTask = new SecondTask();

secondTask.Show();

}

private void Window\_Closing(object sender, System.ComponentModel.CancelEventArgs e)

{

Application.Current.Shutdown();

}

}

}

**Задание 1**

Условие

Разработать приложение с текстовым полем, кнопкой "очистить" и выпадающим списком для задания внешнего вида текстовых полей. Задайте для текстовых поле одинаковый градиентный фон. Кнопка "закрыть" должна быть доступна только в том случае, если в обоих текстовых полях нет текста. Внешний вид текстовых полей (тип шрифта, цвет шрифта) должен меняться в зависимости от значения, выбранного в выпадающем списке.

Код программы

FirstTask.xaml

<Window x:Class="Prac08.FirstTask"

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:Prac08"

mc:Ignorable="d"

Title="FirstTask" Height="450" Width="800" Closing="Window\_Closing">

<Grid>

<TextBox x:Name="textBox1" HorizontalAlignment="Left" Height="23" Margin="190,167,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="120" TextChanged="textBox\_TextChanged"/>

<TextBox x:Name="textBox2" HorizontalAlignment="Center" Height="23" Margin="0,167,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="120" TextChanged="textBox\_TextChanged"/>

<ComboBox x:Name="comboBox1" HorizontalAlignment="Left" Margin="490,167,0,0" VerticalAlignment="Top" Width="120" SelectionChanged="comboBox1\_SelectionChanged">

<ComboBoxItem Content="Arial"/>

<ComboBoxItem Content="Times New Roman"/>

<ComboBoxItem Content="Calibri"/>

</ComboBox>

<Button x:Name="clearButton" Content="Очистить" HorizontalAlignment="Left" Margin="190,217,0,0" VerticalAlignment="Top" Width="120" Click="clearButton\_Click"/>

<Button x:Name="closeButton" Content="Закрыть" HorizontalAlignment="Center" Margin="0,217,0,0" VerticalAlignment="Top" Width="120" IsEnabled="True" Click="closeButton\_Click"/>

<Label Content="Шрифт:" HorizontalAlignment="Left" Margin="526,136,0,0" VerticalAlignment="Top"/>

<Label Content="Текстовое поле 1:" HorizontalAlignment="Left" Margin="196,136,0,0" VerticalAlignment="Top"/>

<Label Content="Текстовое поле 2:" HorizontalAlignment="Center" Margin="0,136,0,0" VerticalAlignment="Top"/>

</Grid>

</Window>

FirstTask.xaml.cs

using System.Windows;

using System.Windows.Controls;

using System.Windows.Media;

namespace Prac08

{

/// <summary>

/// Логика взаимодействия для FirstTask.xaml

/// </summary>

public partial class FirstTask : Window

{

public FirstTask()

{

InitializeComponent();

GradientStopCollection gradientStops = new GradientStopCollection

{

new GradientStop(Colors.LightPink, 0.0),

new GradientStop(Colors.White, 1.0)

};

textBox1.Background = new LinearGradientBrush(gradientStops, new Point(0, 0), new Point(0, 1));

textBox2.Background = new LinearGradientBrush(gradientStops, new Point(0, 0), new Point(0, 1));

}

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

{

textBox1.Clear();

textBox2.Clear();

}

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

{

if (string.IsNullOrWhiteSpace(textBox1.Text) && string.IsNullOrWhiteSpace(textBox2.Text))

{

this.Close();

}

}

private void comboBox1\_SelectionChanged(object sender, SelectionChangedEventArgs e)

{

string fontFamily = ((ComboBoxItem)comboBox1.SelectedItem).Content.ToString();

textBox1.FontFamily = new FontFamily(fontFamily);

textBox2.FontFamily = new FontFamily(fontFamily);

if (fontFamily == "Arial")

{

textBox1.Foreground = Brushes.Blue;

textBox2.Foreground = Brushes.Blue;

}

else if (fontFamily == "Times New Roman")

{

textBox1.Foreground = Brushes.Green;

textBox2.Foreground = Brushes.Green;

}

else if (fontFamily == "Calibri")

{

textBox1.Foreground = Brushes.Red;

textBox2.Foreground = Brushes.Red;

}

}

private void textBox\_TextChanged(object sender, TextChangedEventArgs e)

{

if (string.IsNullOrWhiteSpace(textBox1.Text) && string.IsNullOrWhiteSpace(textBox2.Text))

{

closeButton.IsEnabled = true;

}

else

{

closeButton.IsEnabled = false;

}

}

private void Window\_Closing(object sender, System.ComponentModel.CancelEventArgs e)

{

this.Hide();

MainWindow mainWindow = new MainWindow();

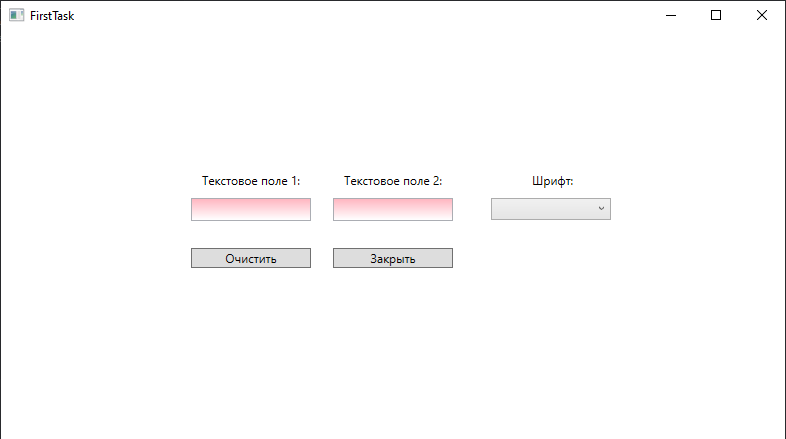
mainWindow.Show();

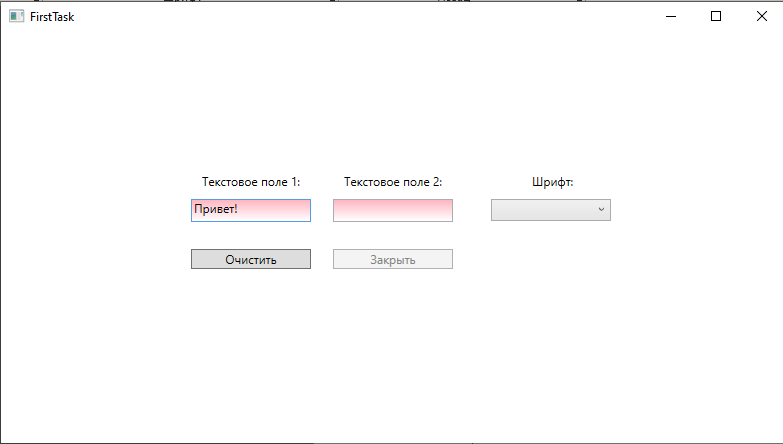
}

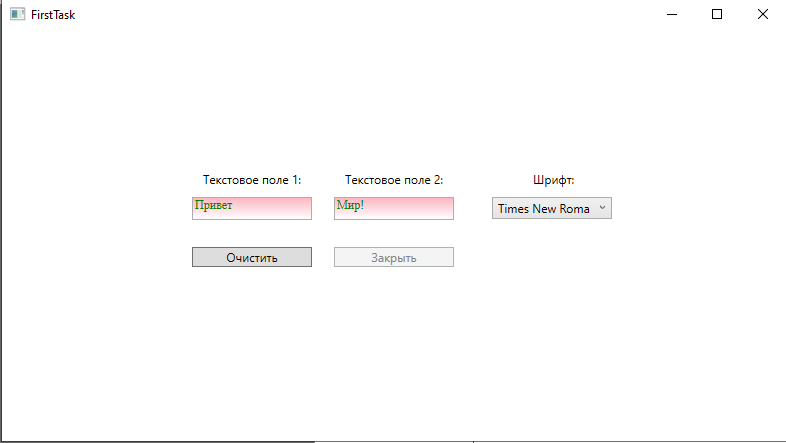
}

}

Результат программы







**Задание 2**

Условие

Разработайте приложение "Пульсар", изображающее квадрат, плавно меняющий своё состояние. Используйте элемент "Rectangle" с радиальной градиентной заливкой RadialGradientBrush для свойства Fill (заливка).

Код программы

SecondTask.xaml

<Window x:Class="Prac08.SecondTask"

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:Prac08"

mc:Ignorable="d"

Title="SecondTask" Height="450" Width="800" Closing="Window\_Closing">

<Grid>

<Rectangle x:Name="Rectangle" Width="400" Height="400" />

</Grid>

</Window>

SecondTask.xaml.cs

using System;

using System.Windows;

using System.Windows.Media;

using System.Windows.Media.Animation;

using System.Windows.Shapes;

namespace Prac08

{

/// <summary>

/// Логика взаимодействия для SecondTask.xaml

/// </summary>

public partial class SecondTask : Window

{

public SecondTask()

{

InitializeComponent();

PulsarRectangleAnimation();

}

private void PulsarRectangleAnimation()

{

RadialGradientBrush brush = new RadialGradientBrush();

brush.GradientStops.Add(new GradientStop(Colors.Yellow, 0.0));

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

GradientStop gradientStop = new GradientStop(Colors.Red, 1.0);

brush.GradientStops.Add(gradientStop);

Rectangle.Fill = brush;

ColorAnimation animation = new ColorAnimation

{

From = Colors.Yellow,

To = Colors.Red,

Duration = new Duration(TimeSpan.FromSeconds(1)),

AutoReverse = true,

RepeatBehavior = RepeatBehavior.Forever

};

gradientStop.BeginAnimation(GradientStop.ColorProperty, animation);

}

private void Window\_Closing(object sender, System.ComponentModel.CancelEventArgs e)

{

this.Hide();

MainWindow mainWindow = new MainWindow();

mainWindow.Show();

}

}

}

Результат программы

