

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»
Филиал
«Минский радиотехнический колледж»

Учебная дисциплина
«Конструирование программ и языка программирования»

Инструкция
по выполнению лабораторной работы №30
«Разработка отладка и испытание программ с шаблонами управления»

Минск 2024 г.

Лабораторная работа № 30

Тема работы: «Разработка отладка и испытание программ с шаблонами управления»

1 Цель работы

Сформировать умения и навыки разработки программ с использованием шаблонов управления.

2 Задание

Видоизменить код, представленный в пункте 5 данной инструкции таким образом, чтобы окно изменило форму.

3 Оснащение работы

ПК, среда Visual Studio 2019, MSword.

4 Основные теоретические сведения

Все визуальные элементы в WPF уже имеют встроенные шаблоны, которые определяют визуальное дерево, структуру и даже поведение элементов. Однако мощь шаблонов состоит в том, что мы можем их переопределить по своему вкусу. Например, сделать круглое окно, а не квадратное, или кнопку в виде морской звезды.

Пример создания округлой кнопки:

```
<Window x:Class="ControlsTemplateApp.MainWindow"
        xmlns="http://schemas.microsoft.com/winfx/2006/
xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/200
6/xaml"
        xmlns:d="http://schemas.microsoft.com/expressio
n/blend/2008"
        xmlns:mc="http://schemas.openxmlformats.org/mar
kup-compatibility/2006"
        xmlns:local="clr-namespace:ControlsTemplateApp"
        mc:Ignorable="d"
        Title="MainWindow" Height="250" Width="300">
    <Window.Resources>
        <ControlTemplate                                TargetType="Button"
x:Key="btTemplate">
            <Border                                    CornerRadius="25"
BorderBrush="CadetBlue" BorderThickness="2"
                Background="LightBlue"    Height="40"
Width="100" >
```

```

        <ContentControl                                Margin="5"
HorizontalAlignment="Center"  VerticalAlignment="Center"
Content="Hello" />
    </Border>
</ControlTemplate>
</Window.Resources>
<Grid>
    <Button                                x:Name="myButton"
Template="{StaticResource    btTemplate}"    Height="40"
Width="100">Hello</Button>
</Grid>
</Window>

```

Результат работы кода представлен на рисунке 37.1

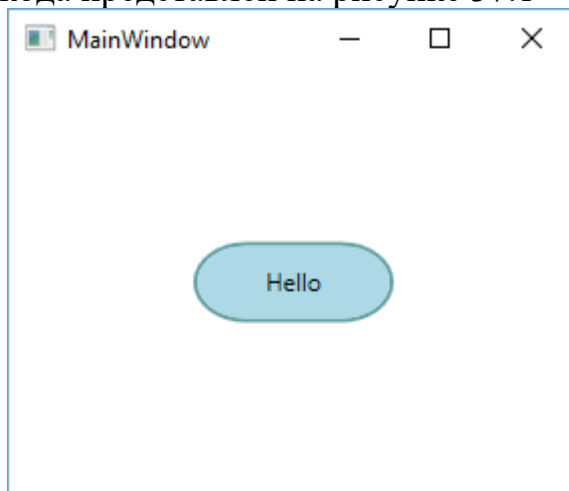


Рисунок 37.1 – Результат работы кода на формирование округлой кнопки

Из элемента, к которому применяется шаблон, можно влиять на свойства, определенные в этом шаблоне. Для этого необходимо использовать элемент **TemplateBinding**. Он служит для установки в шаблоне привязки к свойствам элемента.

Элемент управления обладает определенными состояниями, которые могут влиять на визуальное отображение. Например, для кнопки это может быть состояние **MouseOver**, когда указатель мыши наведен на кнопку, или состояние **Pressed**, когда кнопка нажата. У большинства элементов управления есть несколько состояний.

Чтобы изменить внешний вид в зависимости от состояния элемента используется объект **VisualState**. Объект **VisualState** через другой объект – **Storyboard** позволяет определить анимацию, изменяющую внешний вид элемента.

Анимацией управляет объект **Storyboard**. Его свойство **TargetName** указывает на имя объекта, который анимируется, а **TargetProperty** - свойство объекта, которое анимируется.

Если объект **VisualState** используется для визуализации элемента в определенном состоянии, то объект **VisualTransition** визуализирует элемент в момент перехода от одного состояния в другое.

5. Порядок выполнения работы

Видоизменить следующий код таким образом, чтобы окно имело овальную форму вместо круглой.

```
<Window x:Class="ControlsTemplateApp.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:ControlsTemplateApp"
        mc:Ignorable="d"
        Title="Круглое окно" Height="250"
        Width="300"
        AllowsTransparency="True"
        WindowStyle="None" Background="Transparent">
    <Window.Template>
        <ControlTemplate TargetType="Window">
            <Border Name="newBorder" CornerRadius="150"
                Opacity="0.7" Background="LightBlue">
                <Grid>
                    <Grid.RowDefinitions>
                        <RowDefinition Height="Auto" />
                        <RowDefinition />
                        <RowDefinition Height="Auto" />
                    </Grid.RowDefinitions>
                    <!--Заголовок-->
                    <TextBlock Text="{TemplateBinding
                        Title}" FontWeight="Bold" HorizontalAlignment="Center"
                        MouseLeftButtonDown="TextBlock_
                        MouseLeftButtonDown"/>
                    <!--Основное содержание-->
                    <Border Grid.Row="1">
                        <AdornerDecorator>
                            <ContentPresenter />
                        </AdornerDecorator>
                    </Border>
                    <!--Элемент захвата и изменения
                        размера - работает только для прямоугольных окон-->
                    <ResizeGrip Grid.Row="2"
                        HorizontalAlignment="Right" VerticalAlignment="Bottom"
```

```

Visibility="Collapsed"
IsTabStop="False" />
    </Grid>
    </Border>
</ControlTemplate>
</Window.Template>
<Grid>
    <Button x:Name="closeButton"
Content="Закреть" Click="closeButton_Click"
        Width="80" Height="30"
Background="LightPink" />
    </Grid>
</Window>

```

Файл кода C#:

```

public partial class MainWindow : Window
{
    public MainWindow()
    {
        InitializeComponent();
    }

    private void closeButton_Click(object sender,
RoutedEventArgs e)
    {
        this.Close();
    }

    //Служит для перемещения окна за заголовок
формы
    private void
TextBlock_MouseLeftButtonDown(object sender,
MouseButtonEventArgs e)
    {
        this.DragMove();
    }
}

```

6. Форма отчета о работе

Лабораторная работа № _____

Номер учебной группы _____

Фамилия, инициалы учащегося _____

Дата выполнения работы _____

Тема работы: _____

Цель работы: _____

Оснащение работы: _____
Результат выполнения работы: _____

7. Контрольные вопросы и задания

1. Какой объект управляет анимацией?
2. Перечислите события мыши.

8. Рекомендуемая литература

1. **Рихтер, Дж.** CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C# / Дж. Рихтер. СПб. : Изд-во Питер, 2021. 896 с.
2. **Прайс, М. Дж.** C# 10 и .NET 6. Современная кросс-платформенная разработка / М. Дж. Прайс. СПб : Изд-во Питер, 2023. 848 с.
3. **Васильев, А.Н.** Программирование на C# для начинающих. Особенности языка / А.Н. Васильев. М. : Эксмо, 2022. 528 с.
4. **Фримен, А.** ASP.NET Core 3 с примерами на C# для профессионалов / А. Фримен. СПб. : Изд-во Вильямс, 2021. 1184 с.