**ЛАБОРАТОРНА РОБОТА №1**

**Тема: СТВОРЕННЯ ІНТЕРАКТИВНОЇ КНОПКИ**

**Мета:** Ознайомлення з основами розмітки мови XAML, яка використовується для визначення інтерфейсу користувача в WPF. Створити інтерактивну кнопку в додатку з використанням мови XAML.

**Завдання 14:** Чорного кольору, жирний шрифт (FontWeight), межа: жовта (BorderBrush) з товщиною 1. Кнопка з текстурою. З надписом: "Файли", фон — текстурований, при наведенні відображається легка анімація обертання. Використовуйте прямокутну або квадратну форму для кнопки.

<Window.Resources> визначає глобальний ресурс кольору (TexturedBrush), який можна використовувати в інших елементах через {StaticResource}:

<!-- Визначаємо ресурси -->

<Window.Resources>

<SolidColorBrush x:Key="TexturedBrush" Color="LightGray"/>

</Window.Resources>

Кнопка TexturedButton (150×150) створює із жовтою рамкою, сірим фоном, зображенням та текстом "Файли", розташованими вертикально всередині StackPanel:

<Grid>

<Button x:Name="TexturedButton"

Width="150" Height="150"

FontWeight="Bold"

BorderBrush="Yellow"

BorderThickness="1"

Background="LightGray">

<StackPanel Orientation="Vertical" HorizontalAlignment="Center">

<Image Source="Image/texture.jpg"

Width="40" Height="40" Margin="0,0,0,5"/>

<TextBlock Text="Файли" Foreground="Black" FontSize="18" FontWeight="Bold"

HorizontalAlignment="Center"/>

</StackPanel>

Код додає анімацію обертання для кнопки: при наведенні (`MouseEnter`) вона безкінечно крутиться, а при відведенні (`MouseLeave`) швидко зупиняється, використовуючи `RotateTransform`:

<!-- Тригери для керування анімацією -->

<Button.Triggers>

<!-- При наведеннізапускаємо анімацію -->

<EventTrigger RoutedEvent="MouseEnter">

<BeginStoryboard>

<Storyboard>

<DoubleAnimation Storyboard.TargetName="rotateTransform"

Storyboard.TargetProperty="Angle"

By="360"

Duration="0:0:1"

RepeatBehavior="Forever"/>

</Storyboard>

</BeginStoryboard>

</EventTrigger>

<!-- При відході курсора зупиняємо анімацію -->

<EventTrigger RoutedEvent="MouseLeave">

<BeginStoryboard>

<Storyboard>

<DoubleAnimation Storyboard.TargetName="rotateTransform"

Storyboard.TargetProperty="Angle"

To="{Binding (UIElement.RenderTransform).(RotateTransform.Angle), RelativeSource={RelativeSource Self}}"

Duration="0:0:0.1"/>

</Storyboard>

</BeginStoryboard>

</EventTrigger>

</Button.Triggers>

<Button.RenderTransform>

<RotateTransform x:Name="rotateTransform" Angle="0"/>

</Button.RenderTransform>

<Button.RenderTransformOrigin>0.5,0.5</Button.RenderTransformOrigin>

</Button>

</Grid>

Код створює анімацію обертання кнопки TexturedButton: при наведенні (MouseEnter) запускається безкінечне обертання, а при відведенні (MouseLeave) воно зупиняється:

private void SetupAnimation()

{

rotateStoryboard = new Storyboard();

DoubleAnimation rotateAnimation = new DoubleAnimation

{

By = 360,

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

RepeatBehavior = RepeatBehavior.Forever

};

Storyboard.SetTarget(rotateAnimation, TexturedButton.RenderTransform);

Storyboard.SetTargetProperty(rotateAnimation, new PropertyPath("Angle"));

rotateStoryboard.Children.Add(rotateAnimation);

}

private void TexturedButton\_MouseEnter(object sender, System.Windows.Input.MouseEventArgs e)

{

rotateStoryboard.Begin();

}

private void TexturedButton\_MouseLeave(object sender, System.Windows.Input.MouseEventArgs e)

{

rotateStoryboard.Stop();

}

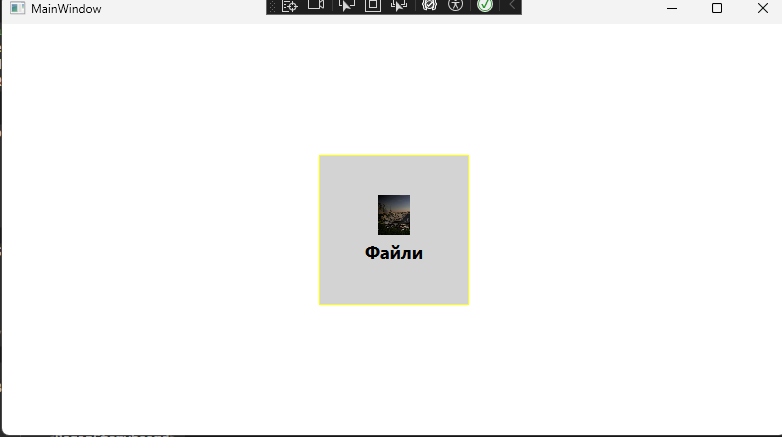


Рис.1. Інтерактивна кнопка

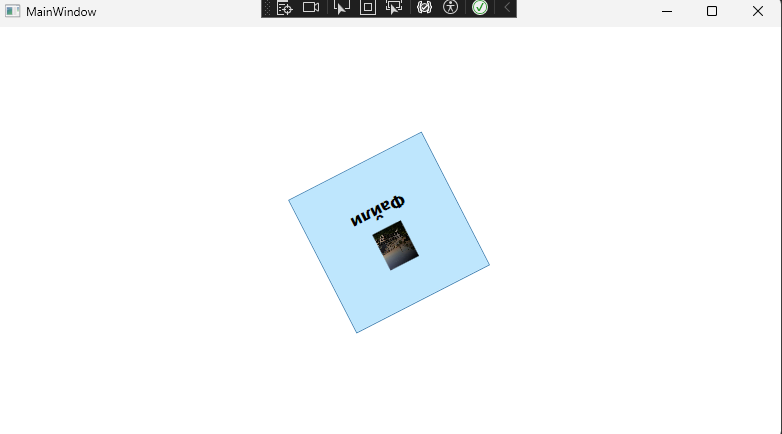


Рис.2. Анімація кнопки при наведенні

**Висновок:** Було створено інтерактивну кнопку в XAML з анімацією обертання, що допомогло ознайомитися з основами розмітки та взаємодії в WPF.

**Виконано:** Куліш Владою Іванівною, група БІП-1-22.