**ПРАКТИЧЕСКОЕ ЗАДАНИЕ 25**

**ТЕХНОЛОГИЯ WINDOWS PRESENTATION FOUNDATION**

Задание 1. Примеры создания WPF приложения. Переделать задание 1 таким образом чтобы главная форма приняла следующий вид и все элементы выполняли необходимые действия.

Анализ результатов:

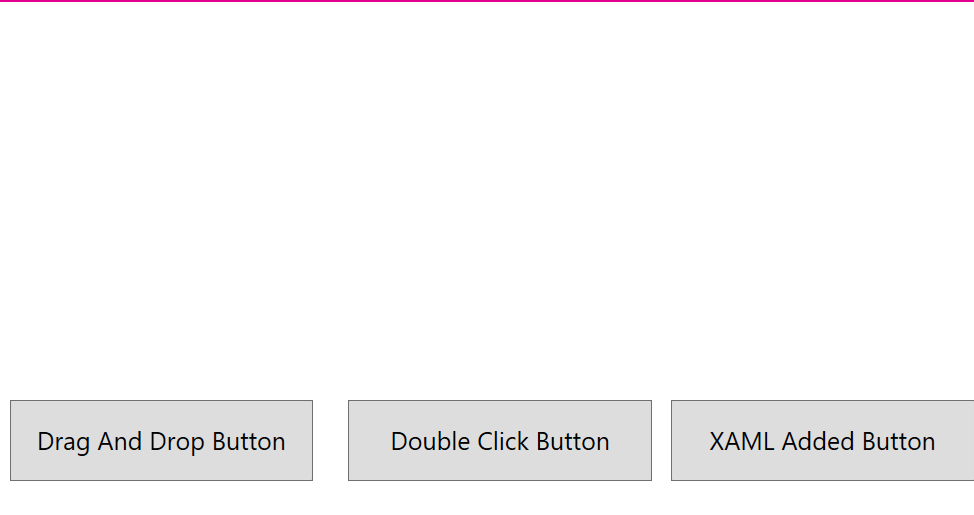


Рисунок 1.1 – Результат работы программы

Источник: собственная разработка

Листинг программы:

public MainWindow()

{

InitializeComponent();

ButtonRun.Click += ButtonRun\_Click\_1;

ButtonAbout.Click += ButtonAbout\_Click;

}

private void ButtonRun\_Click\_1(object sender, RoutedEventArgs e)

{

String hello = "Hello, ";

String input = string.Empty;

if (string.IsNullOrEmpty(TextBoxEnterName.Text) || string.IsNullOrWhiteSpace(TextBoxEnterName.Text))

{

input = "World!";

}

else

{

input = TextBoxEnterName.Text;

}

TextBlockHello.Text = $"{hello} {input}";

}

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

{

string info = $"About program{this.Title}";

AboutApp aboutApp = new AboutApp(info);

aboutApp.ShowDialog();

}

Анализ результатов:

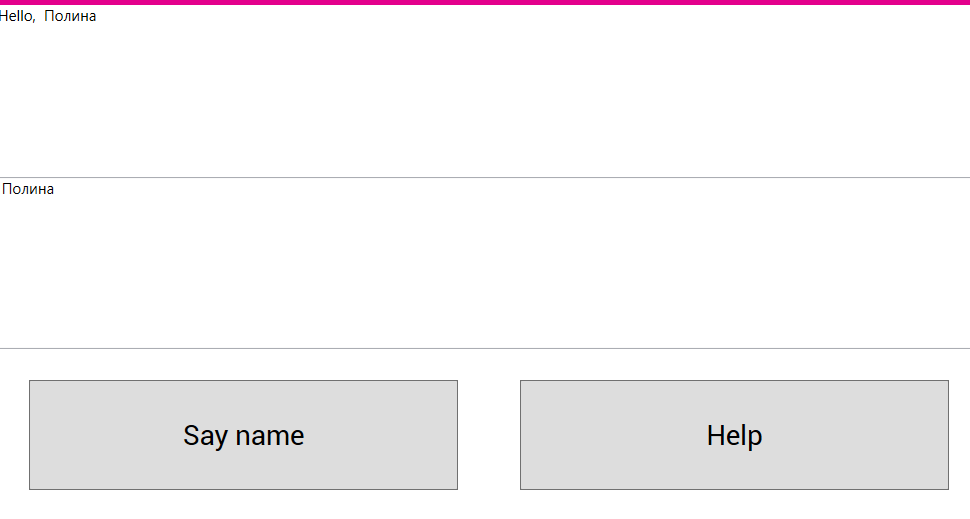


Рисунок 1.2 – Результат работы программы

Источник: собственная разработка

Задание 2. Переделать задание 1 таким образом чтобы главная форма приняла следующий вид и все элементы выполняли необходимые действия.

Листинг программы:

public MainWindow()

{

InitializeComponent();

}

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

{

this.Close();

}

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

{

}

private void TextBox1\_GotFocus(object sender, RoutedEventArgs e)

{

if (TextBox1.Text == "Введите имя")

{

TextBox1.Text = "";

}

}

private void Button\_Click\_1(object sender, RoutedEventArgs e)

{

LabelName.Text = "Hello " + TextBox1.Text;

}

private void Button\_Click\_2(object sender, RoutedEventArgs e)

{

MessageBox.Show("Enter your name and click button 'Скачать Имя'");

} TextBlockHello.Text = $"{hello} {input}";

}

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

{

string info = $"About program{this.Title}";

AboutApp aboutApp = new AboutApp(info);

aboutApp.ShowDialog();

}

Анализ результатов:

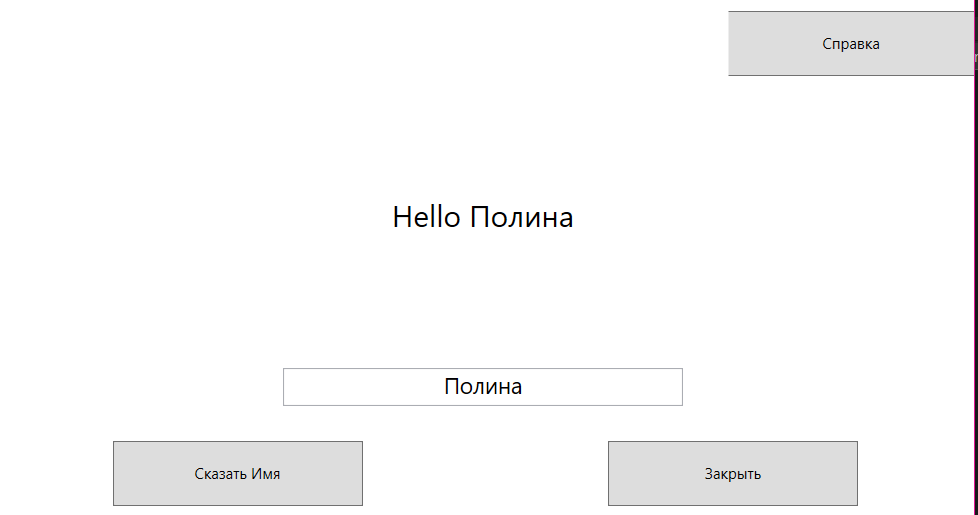


Рисунок 1.3 – Результат работы программы

Источник: собственная разработка

Задание 3. Создайте проект WPF и выполните решение следующего алгоритма. Элементы управления Button TextBlock, TextBox. Расположение элементов продумайте самостоятельно. sinx\*cosx

Листинг программы:

public MainWindow()

{

InitializeComponent();

buttonRun.Click += ButtonRun\_Click;

buttonAbout.Click += ButtonAbout\_Click;

}

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

{

string hello = "HELLO";

string input = string.Empty;

if (string.IsNullOrEmpty(textBoxEnterName.Text) || string.IsNullOrWhiteSpace(textBoxEnterName.Text))

{

input = "WORLD!";

}

else

{

input = textBoxEnterName.Text;

}

textBlockHello.Text = $"{hello} {input}";

}

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

{

string info = $"О программе {this.Title}";

AboutAp aboutAp = new AboutAp(info);

aboutAp.ShowDialog();

}

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

{

this.Close();

}

Анализ результатов:

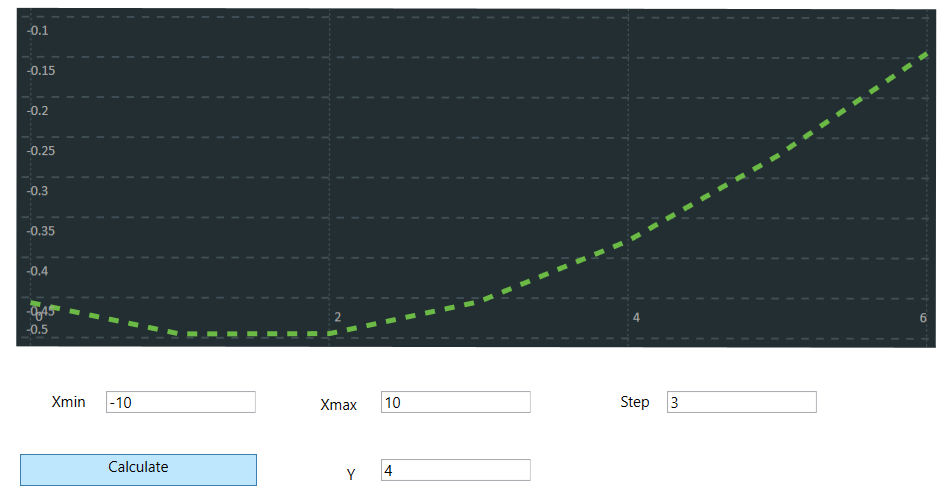


Рисунок 1.4 – Результат работы программы

Источник: собственная разработка

Задание 4. Написать программу с кнопкой. При наведении курсора на элемент управления он должен создавать новую кнопку, а при клике удалять себя.

Листинг программы:

public partial class MainWindow : Window

{

Random Random = new Random();

private int count = 0;

public MainWindow()

{

InitializeComponent();

}

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

{

Button btn = new Button();

btn.MouseEnter += MouseEnter;

btn.Click += Button\_Click;

btn.Content = "Дочерняя кнопка " + Convert.ToString(count);

btn.Margin = new Thickness(Random.Next(0, 430), Random.Next(0, 400), Random.Next(0, 430), Random.Next(0, 500));

form1.Children.Add(btn);

count++;

}

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

{

Button button = (Button)sender;

form1.Children.Remove(button);

}

private void MouseEnter(object sender, MouseEventArgs e)

{

Button button = new Button();

button.MouseEnter += MouseEnter;

button.Click += Button\_Click;

button.Content = "Дочерняя кнопка дочерней кнопки";

button.Margin = new Thickness(Random.Next(0, 430), Random.Next(0, 400), Random.Next(0, 430), Random.Next(0, 500));

form1.Children.Add(button);

}

Анализ результатов:



Рисунок 1.5 – Результат работы программы

Источник: собственная разработка