Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

Тамбовский государственный технический университет

Кафедра

Отчёт по лабораторной работе №1

по дисциплине «ООП»

Выполнил студент гр.

.

Проверил:.

Тамбов 20

***Цели и задачи****.*

Взять готовый компонент и, используя механизм наследования, а также полиморфизма, создать производный класс. В новом классе добавить 2 новых свойства и 1 метод. Использовать не менее 5 экземпляров полученного класса.

***Теория.***

Наследование является одним из трех основополагающих принципов объектно-ориентированного программирования, поскольку оно допускает создание иерархических классификаций. Благодаря наследованию можно создать общий класс, в котором определяются характерные особенности, присущие множеству связанных элементов. От этого класса могут затем наследовать другие, более конкретные классы, добавляя в него свои индивидуальные особенности.

В языке С# класс, который наследуется, называется базовым, а класс, который наследует, — производным. Следовательно, производный класс представляет собой специализированный вариант базового класса. Он наследует все переменные, методы, свойства и индексаторы, определяемые в базовом классе, добавляя к ним свои собственные элементы.

Ниже приведена общая форма объявления класса, наследующего от базового класса.

class имя\_производного\_класса **:** имя\_базового\_класса {

// тело класса

}

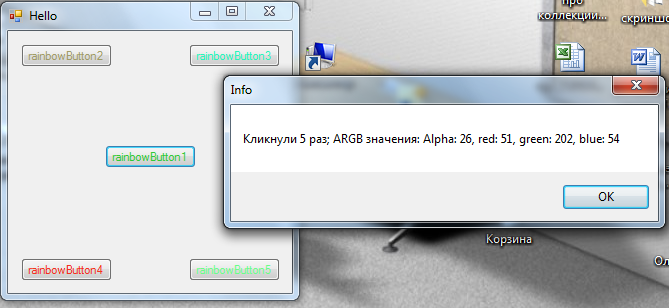
***Описание.***

Создадим класс RainbowButton, который является производным от класса Button. В получившийся класс добавим новый метод ChangeTextColor, задающий случайный цвет текста, а также свойства r, g, b, a (отражают цвет текста в ARGB) и свойство numberOfClicks, показывающее, сколько раз была нажата кнопка.

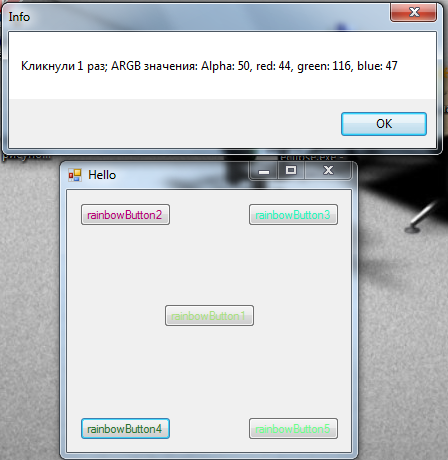
При нажатии на любую из кнопок она меняет свой цвет на случайный, а также вызывает окно MessageBox, в котором написано количество произведенных нажатий на эту кнопку и заданный на ней цвет в формате ARGB.

***Пример работы.***

Кликнули на кнопку rainbowButton1 5 раз:



Кликнули на кнопку rainbowButton4 первый раз:



***Листинг.***

// Program.cs

using System;

using System.Windows.Forms;

namespace NewProject

{

static class Program

{

/// <summary>

/// Главная точка входа для приложения.

/// </summary>

[STAThread]

static void Main()

{

Application.EnableVisualStyles();

Application.SetCompatibleTextRenderingDefault(false);

Application.Run(new Form1());

}

}

}

// Form1.cs

using System;

using System.Drawing;

using System.Windows.Forms;

namespace NewProject

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

}

public class RainbowButton : Button // Наследуемся от класса кнопки

{

private Random rand;

public byte r, g, b, a;

public int numberOfClicks;

public RainbowButton() // Конструктор класса

{

ChangeTextColor();

r = this.ForeColor.R;

g = this.ForeColor.G;

b = this.ForeColor.B;

a = this.ForeColor.A;

this.numberOfClicks=0;

}

public void ChangeTextColor()

{

rand = new Random(); // инициализируем рандомизатор

this.ForeColor = Color.FromArgb(rand.Next(int.MaxValue)); // задаем случайный цвет текста

}

protected override void OnMouseClick(MouseEventArgs e) // переопределяем обработчик базового класса

// Данный метод вызывается при клике по кнопке

{

ChangeTextColor();

base.OnMouseClick(e); // вызываем обработчик базового класса, что бы ничего не "потерялось"

this.numberOfClicks++;

r = this.ForeColor.R;

g = this.ForeColor.G;

b = this.ForeColor.B;

a = this.ForeColor.A;

string text = String.Format("ARGB значения: Alpha: {0}, " +

"red: {1}, green: {2}, blue: {3}", new object[] { a, r, g, b });

MessageBox.Show("Кликнули " + this.numberOfClicks + " раз; "+ text, "Info");

}

}

}