

Разработка программного приложения с пользовательским интерфейсом в C#. Основные элементы

Языки и методы
программирования

План

лекции

- Среда Visual Studio 2019: формы, основные интерфейсные и служебные элементы. Свойства и события элементов.
- Демонстрация работы с компонентами на языке C# в Visual Studio 2019.

Среда Visual Studio 2019:
формы, основные
интерфейсные и служебные
элементы.

Свойства и события элементов

Форма

Windows Forms - интерфейс программирования приложений (API), отвечающий за графический интерфейс пользователя и являющийся частью Microsoft .NET Framework.

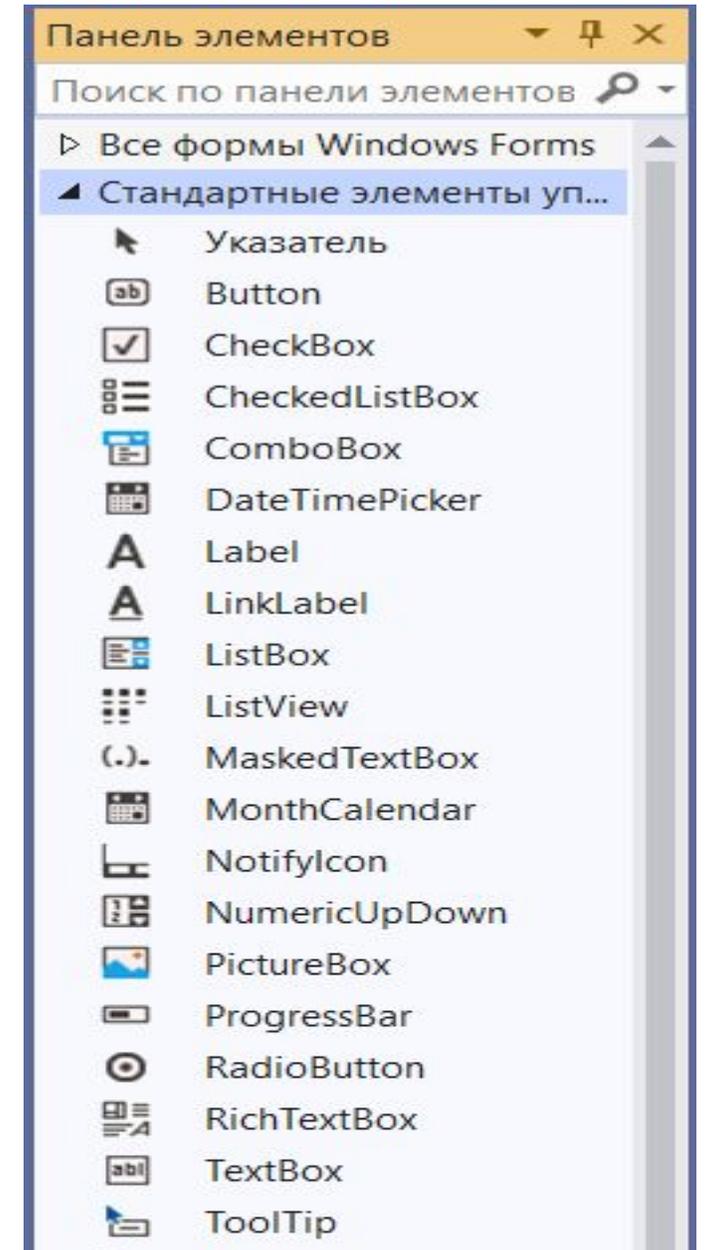


Название	Описание
Text	Пользовательское название формы
Visible	Получает или задает значение, указывающее, отображается ли элемент управления и все его дочерние элементы управления.
WindowState	Возвращает или задает значение, указывающее на состояние формы: развернутое, свернутое или обычное.
FormBorderStyle	Получает или задает стиль границы формы. Может принимать значения: None – форма не имеет границ; FixedSingle – обычная форма без возможности масштабирования; Sizable – обычная форма с возможностью масштабирования.
MinimizeBox	Возвращает или задает значение, указывающее, отображается ли кнопка свертывания в строке заголовка формы.
MaximizeBox	Возвращает или задает значение, указывающее, отображается ли кнопка развертывания в строке заголовка формы.

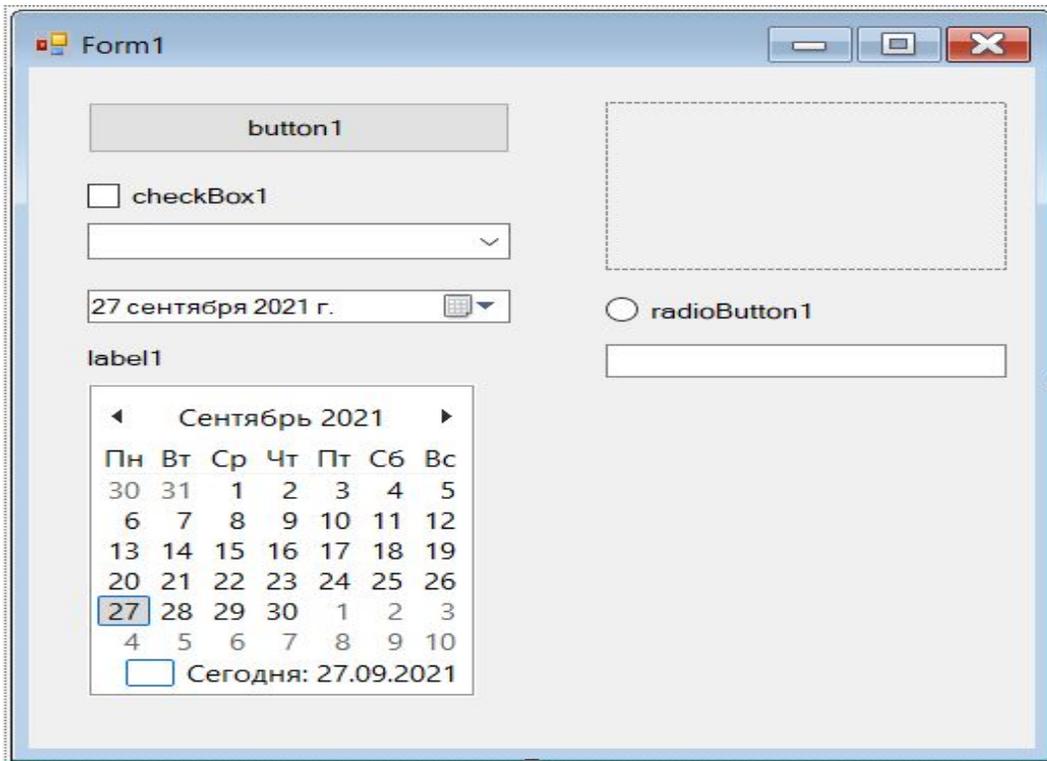
Панель элементов

Windows Forms предоставляет широкий спектр элементов, которые условно можно разделить на два типа: интерфейсные – те которые видны пользователю и с которыми он может работать непосредственно (разного рода кнопки, панели, таблицы) и служебные – те, что выполняют определенные задачи и вызываются путем взаимодействия пользователя с интерфейсными элементами (диалоги, таймеры, адаптеры).

Перед использованием элемент должен быть помещен на форму (обычно при помощи drag&drop или как его называют «перетаскивание»), за исключением самого элемента формы (элемент форма создается автоматически при создании проекта).



Интерфейсные элементы



The screenshot shows a Windows form titled "Form1" with the following elements:

- A button labeled "button1".
- An unchecked checkbox labeled "checkBox1".
- A dropdown menu.
- A date picker showing "27 сентября 2021 г." with a calendar icon.
- A label labeled "label1" containing a calendar for September 2021. The date 27 is highlighted.
- A radio button labeled "radioButton1".
- A text input field.

Сентябрь 2021						
Пн	Вт	Ср	Чт	Пт	Сб	Вс
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	1	2	3
4	5	6	7	8	9	10

Сегодня: 27.09.2021

- Кнопка (button)
- «Флажок» (checkBox)
- Текстовое поле (TextBox)
- Метка (Label)
- CheckBox
- ...

Кнопка (Button)



Название	Описание
Text	Получает или задает текст, сопоставленный с этим элементом управления.
TextAlign	Получает или задает выравнивание текста в элементе управления " Кнопка " .
Image	Возвращает или задает изображение, отображаемое на элементе управления "Кнопка".
Visible	Получает или задает значение, указывающее, отображается ли элемент управления.
Enabled	Возвращает или задает значение, показывающее, сможет ли элемент управления отвечать на действия пользователя.
Name	Возвращает или задает имя элемента управления. Стандарты те же что и для переменных C++.

Название	Описание
Click	Происходит при щелчке элемента управления.
DoubleClick	Происходит при двойном щелчке мышью элемента управления Button.

Текстовое поле (TextBox)



Элемент управления текстовым окном для ввода данных пользователем.

Название	Описание
Text	Получает или задает текст, сопоставленный с этим элементом управления.
Multiline	Определяет поддерживает ли элемент многострочный текст.
Name	Возвращает или задает имя элемента управления. Стандарты те же что и для переменных C++.
ReadOnly	Возвращает или задает значение, определяющее возможность изменения содержимого элемента управления TextBox.
Rows	Возвращает или задает число строк, отображаемых в многострочном текстовом окне.

Название	Описание
Click	Происходит при щелчке элемента управления.
DoubleClick	Происходит при двойном щелчке мышью элемента управления TextBox.
KeyDown	Происходит при нажатии клавиши на клавиатуре, когда фокус находится на элементе TextBox.
TextChanged	Происходит при изменении содержимого текстового окна TextBox.

Метка (Label)



Представляет стандартную надпись Windows. Находится в группе элементов Common Controls.

Название	Описание
Text	Получает или задает текст, сопоставленный с этим элементом управления.
TextAlign	Получает или задает выравнивание текста в элементе управления Label.
Image	Возвращает или задает изображение, отображаемое на элементе управления Label.
Visible	Получает или задает значение, указывающее, отображается ли элемент управления.
Enabled	Возвращает или задает значение, показывающее, сможет ли элемент управления отвечать на действия пользователя.
Name	Возвращает или задает имя элемента управления. Стандарты те же что и для переменных C++.

Название	Описание
Click	Происходит при щелчке элемента управления.
DoubleClick	Происходит при двойном щелчке мышью элемента управления Button.

«Флажок» (RadioButton)



Представляет переключатель, который пользователь может устанавливать (выбирать), но не снимать (отменять выбор). Находится в группе элементов Common Controls

Название	Описание
Text	Получает или задает текст, сопоставленный с этим элементом управления.
Visible	Получает или задает значение, указывающее, отображается ли элемент управления.
Enabled	Возвращает или задает значение, показывающее, сможет ли элемент управления отвечать на действия пользователя.
Name	Возвращает или задает имя элемента управления.
Checked	Возвращает или задает значение, указывающее, находится ли RadioButton во включенном состоянии.

Название	Описание
Click	Происходит при щелчке элемента управления.
DoubleClick	Происходит при двойном щелчке мышью элемента управления RadioButton.
CheckedChanged	Возникает при переходе RadioButton во включённое состояние.

CheckBox

Представляет элемент управления - флажок, который пользователь может устанавливать и снимать. Находится в группе элементов Common Controls.

Название	Описание
Text	Получает или задает текст, сопоставленный с этим элементом управления.
Visible	Получает или задает значение, указывающее, отображается ли элемент управления.
Enabled	Возвращает или задает значение, показывающее, сможет ли элемент управления отвечать на действия пользователя.
Name	Возвращает или задает имя элемента управления. Стандарты те же что и для переменных C++.
Checked	Возвращает или задает значение, указывающее, находится ли CheckBox во включенном состоянии.
CheckState	Возвращает или задает состояние CheckBox. Помимо отмеченного и пустого может принимать еще и всегда отмеченное состояние.

Название	Описание
Click	Происходит при щелчке элемента управления.
DoubleClick	Происходит при двойном щелчке мышью элемента управления CheckBox.
CheckedChanged	Возникает при переходе RadioButton во включенное состояние.

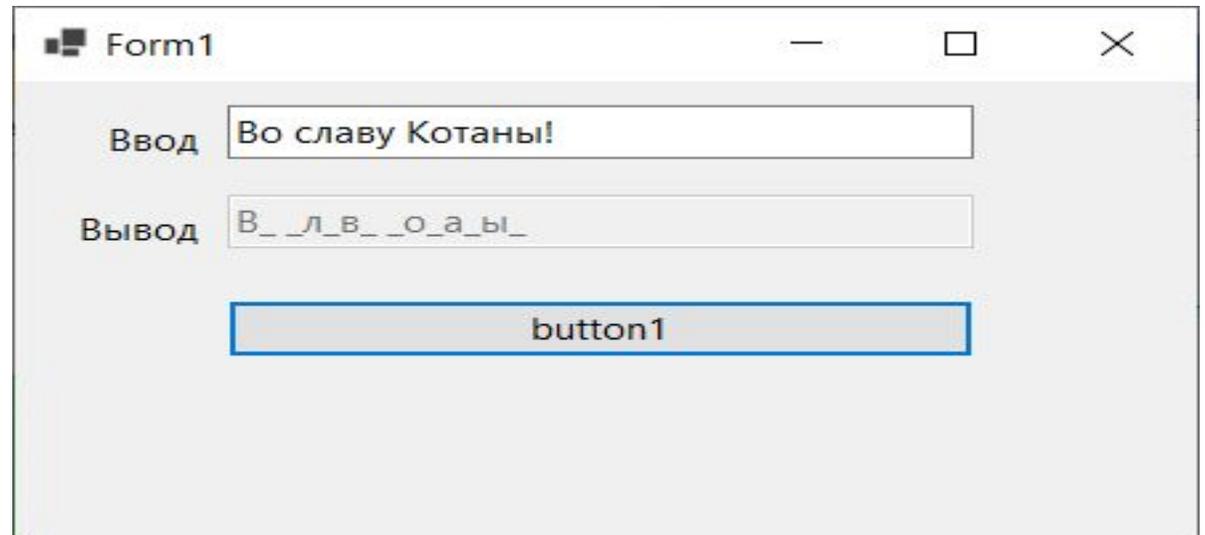
Демонстрация работы с
компонентами на языке C# в
Visual Studio 2019.



TextBox

Задание. Организовать для текста в TextBox обработку по следующему алгоритму: заменить каждый второй символ на знак подчеркивания

```
private void button1_Click(object sender, EventArgs e)
{
    string str = textBox1.Text;
    string str2 = "";
    for (int i=0; i<str.Length; i++)
    {
        if (i%2==0) {
            str2 = str2 + str[i];
        }
        else
        {
            str2 = str2 + "_";
        }
    }
    textBox2.Text = str2;
}
```

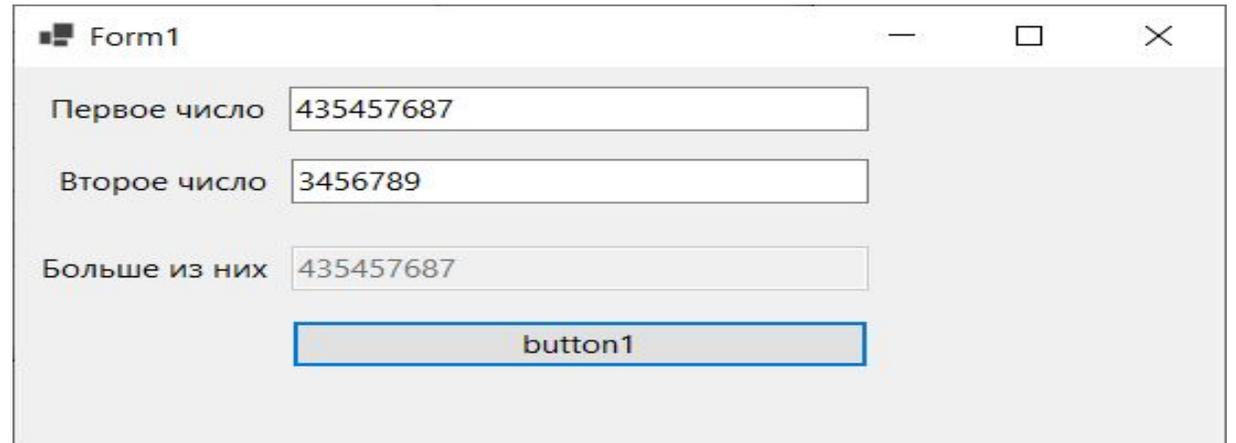


TextBox

Задание. Организовать для двух чисел в TextBox вывод большего из них в третий TextBox.

```
public int Sravn(int a, int b)
{
    if (a>b) {
        return a;
    }
    if (b>a) {
        return b;
    }
}

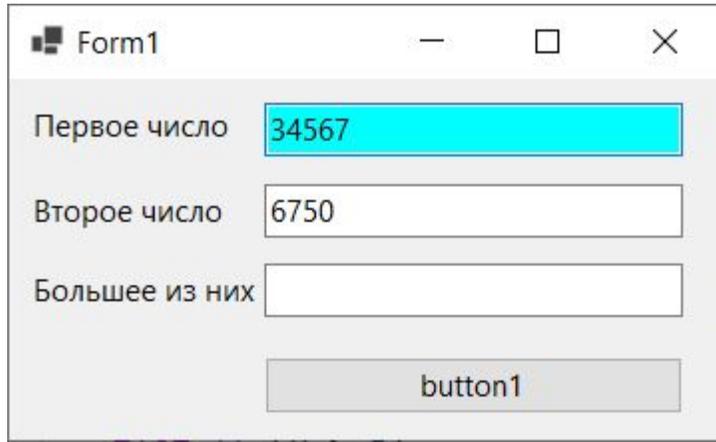
private void button1_Click(object sender, EventArgs e)
{
    int x = Convert.ToInt32(textBox1.Text);
    int y = Convert.ToInt32(textBox3.Text);
    textBox2.Text = Convert.ToString(Sravn(x,y));
}
```



The screenshot shows a Windows application window titled "Form1". It contains three text boxes and a button. The first text box, labeled "Первое число", contains the value "435457687". The second text box, labeled "Второе число", contains the value "3456789". The third text box, labeled "Больше из них", contains the value "435457687". Below the third text box is a button labeled "button1".

TextBox

Задание. Организовать для двух чисел в TextBox подсвечивание большего из них.



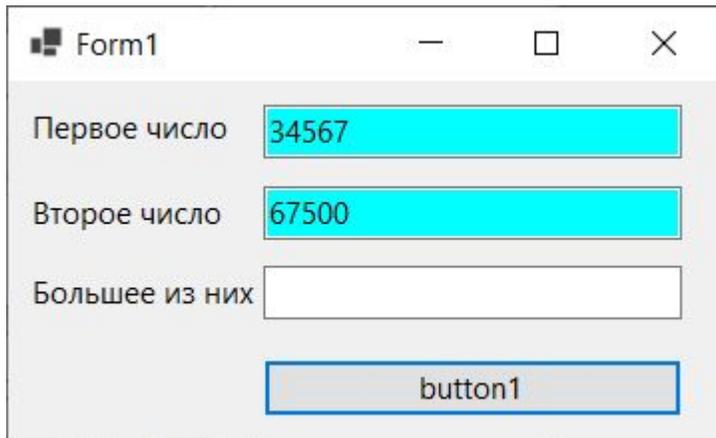
Form1

Первое число 34567

Второе число 6750

Большее из них

button1



Form1

Первое число 34567

Второе число 67500

Большее из них

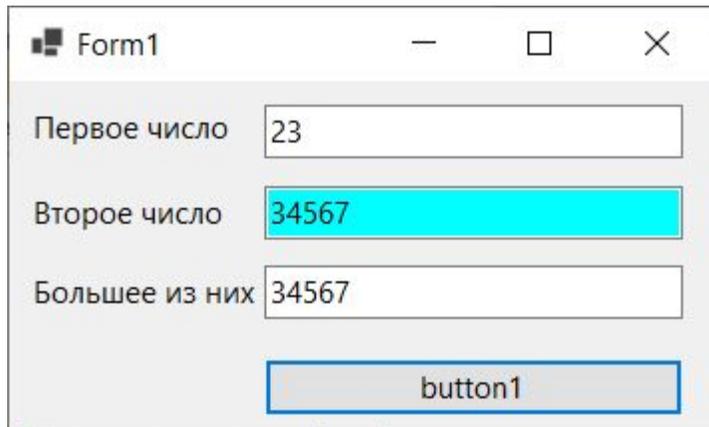
button1

```
private void button1_Click(object sender, EventArgs e)
{
    int a = Convert.ToInt32(textBox1.Text);
    int b = Convert.ToInt32(textBox2.Text);
    if (b > a) {
        textBox2.BackColor = Color.Aqua;
    }
    else if (b < a)
    {
        textBox1.BackColor = Color.Aqua;
    }
}
```

При запуске программы textBox с большим из чисел подсвечивается. Однако при изменении чисел программа «ломается», поскольку предыдущее подсвечивание не отменяется.

TextBox

Задание. Организовать для двух чисел в TextBox подсвечивание большего из них.



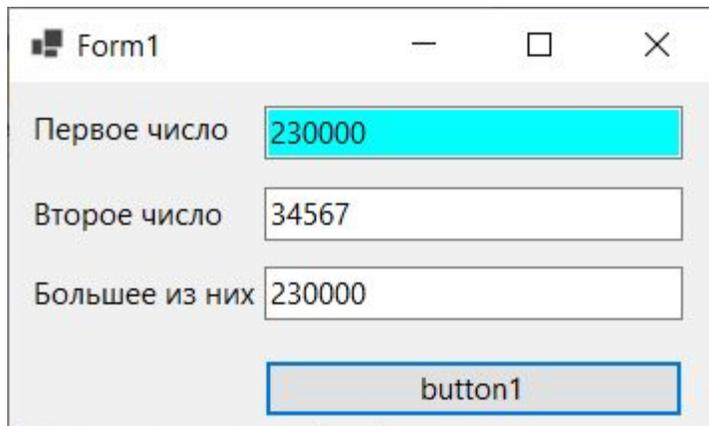
Form1

Первое число 23

Второе число 34567

Большее из них 34567

button1



Form1

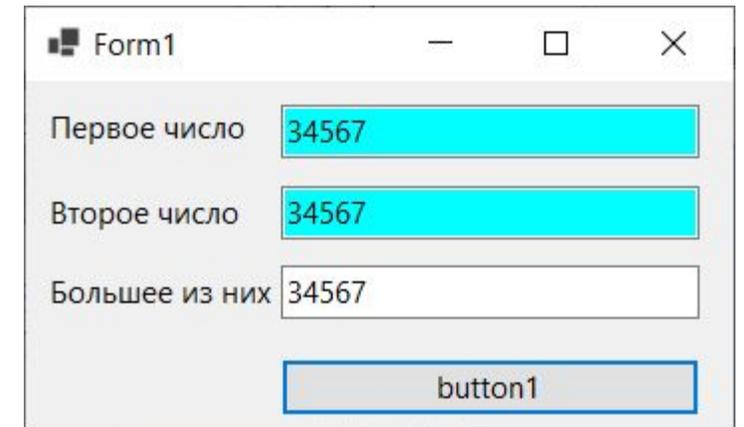
Первое число 230000

Второе число 34567

Большее из них 230000

button1

```
private void button1_Click(object sender, EventArgs e)
{
    int a = Convert.ToInt32(textBox1.Text);
    int b = Convert.ToInt32(textBox2.Text);
    if (b > a) {
        textBox1.BackColor = Color.White;
        textBox2.BackColor = Color.Aqua;
    }
    else if (b < a)
    {
        textBox2.BackColor = Color.White;
        textBox1.BackColor = Color.Aqua;
    }
    else
    {
        textBox1.BackColor = Color.Aqua;
        textBox2.BackColor = Color.Aqua;
    }
}
```



Form1

Первое число 34567

Второе число 34567

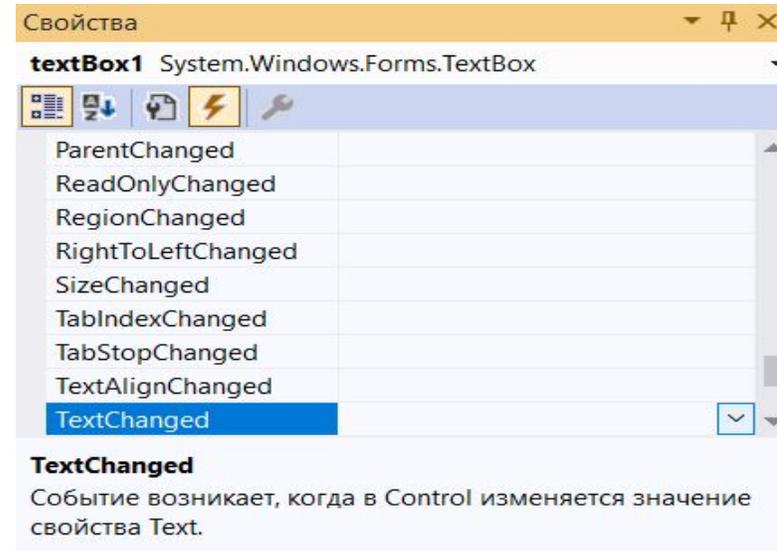
Большее из них 34567

button1

TextBox

Задание. Организовать для двух чисел в TextBox подсвечивание большего из них. Настроить TextBox таким образом, чтобы результат пересчитывался при изменении чисел в любом из TextBox, а не только при нажатии кнопки.

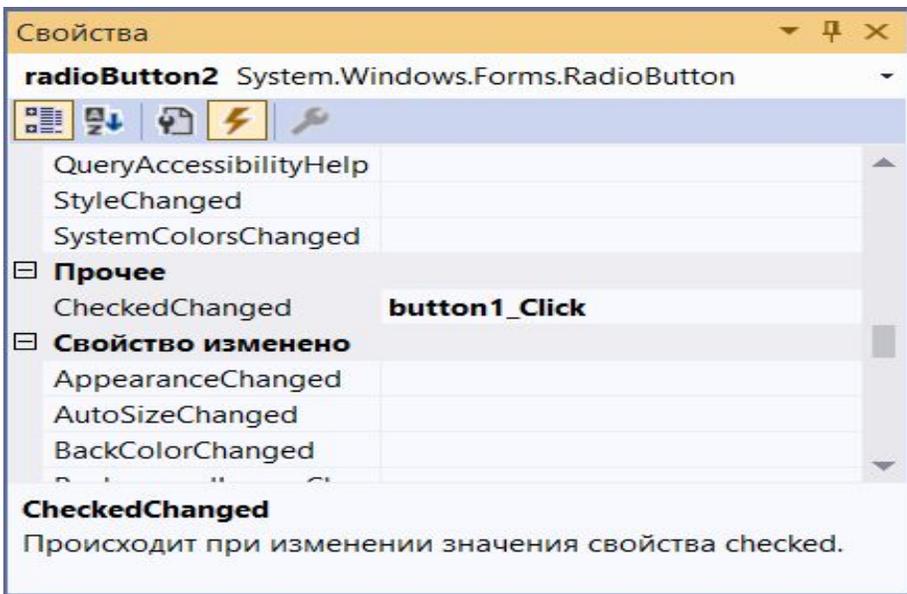
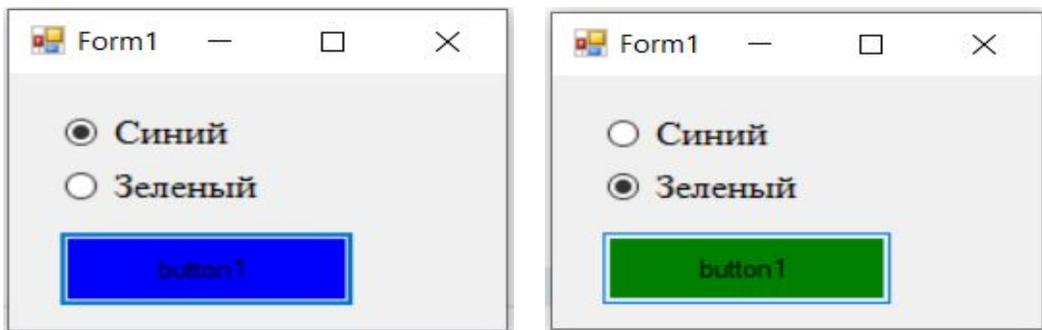
```
public void Sravn()
{
    int a = Convert.ToInt32(textBox1.Text);
    int b = Convert.ToInt32(textBox2.Text);
    if (a > b)
    {
        textBox2.BackColor = Color.White;
        textBox1.BackColor = Color.Aqua;
    }
    else if (b > a)
    {
        textBox1.BackColor = Color.White;
        textBox2.BackColor = Color.Aqua;
    }
    else {
        textBox1.BackColor = Color.Aqua;
        textBox2.BackColor = Color.Aqua;
    }
}
```



```
private void button1_Click(object sender, EventArgs e){
    Sravn();
}
private void textBox1_TextChanged(object sender, EventArgs e){
    Sravn();
}
private void textBox2_TextChanged(object sender, EventArgs e){
    Sravn();
}
```

Задание. Организовать обработку выбора цвета кнопки через элемент RadioButton.

RadioButton

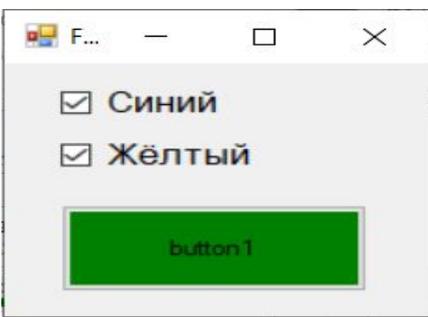
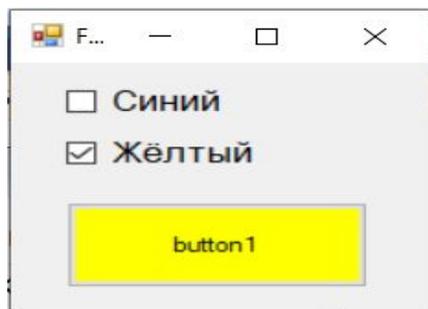
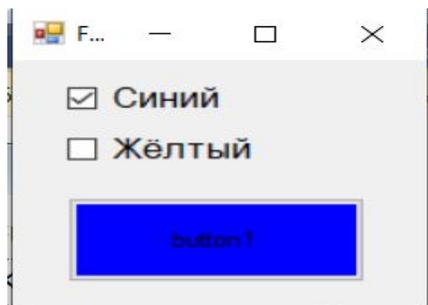


```
private void button1_Click(object sender, EventArgs e)
{
    if (radioButton1.Checked == true){
        button1.BackColor = Color.Blue; }
    if (radioButton2.Checked == true){
        button1.BackColor = Color.Green;}
}
```

```
public void Painting()
{
    if (radioButton1.Checked == true)
        {button1.BackColor = Color.Blue;}
    if (radioButton2.Checked == true)
        {button1.BackColor = Color.Green;}
}
private void button1_Click(object sender, EventArgs e)
{    Painting();
}
```

CheckBox

Задание. Организовать обработку выбора цвета кнопки через элемент CheckBox.



```
public void Painting()
{
    button1.BackColor = Color.White;
    if (checkBox1.Checked == true)
        {button1.BackColor = Color.Blue;}
    if (checkBox2.Checked == true)
        {button1.BackColor = Color.Yellow;}
    if ((checkBox1.Checked == true) && (checkBox2.Checked == true))
        {button1.BackColor = Color.Green;}
}
private void button1_Click(object sender, EventArgs e)
{
    Painting();
}
```

Задание. Организовать добавление текста из listBox через элемент textBox.

ListBox

Form1

Введите текст:

- кот
- хамелеон
- крокодил
- обезьяна
- утка
- ХОМЯК

```
private void button1_Click(object sender, EventArgs e)
{
    listBox1.Items.Add(textBox1.Text);
}
```

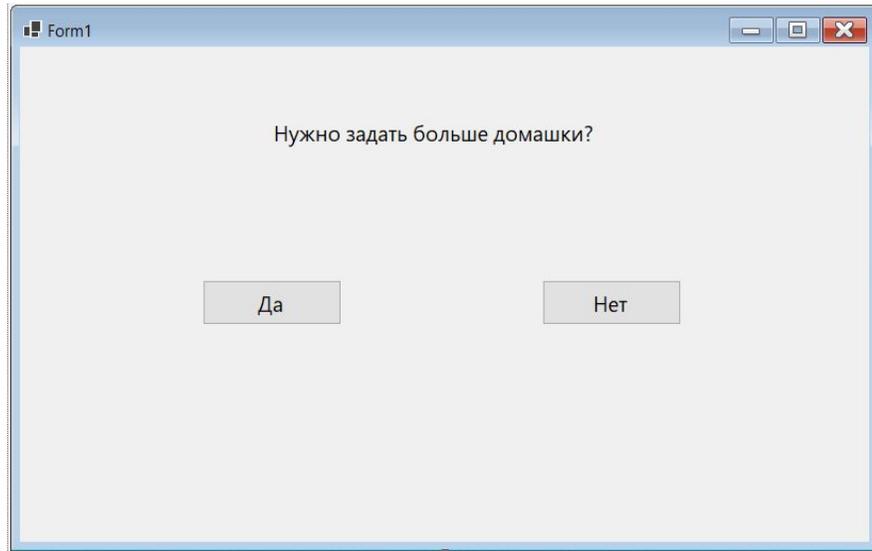
Form1

Введите текст:


```
private void button2_Click(object sender, EventArgs e)
{
    listBox1.Items.Clear();
}
```

Button

Задание. Реализовать программу: на форме в метке Label находится какой-либо вопрос, есть 2 кнопки с разными ответами, одна кнопка статична, другая «убегает» от указателя мыши.



Отступ нужно организовать так, чтобы кнопка не «вылетала» за видимую часть формы, поэтому можно взять генератор случайных чисел и задать диапазон от 0 до ширины формы.

При этом если случайное число приближается к правой и нижней границам формы – кнопка будет частично невидима, поэтому следует из характеристик формы (Width, Height) вычесть соответствующие характеристики кнопки.

Для решения данной задачи потребуется событие, возникающее, когда указатель мыши попадает на кнопку. Таким событием является **MouseEnter**.

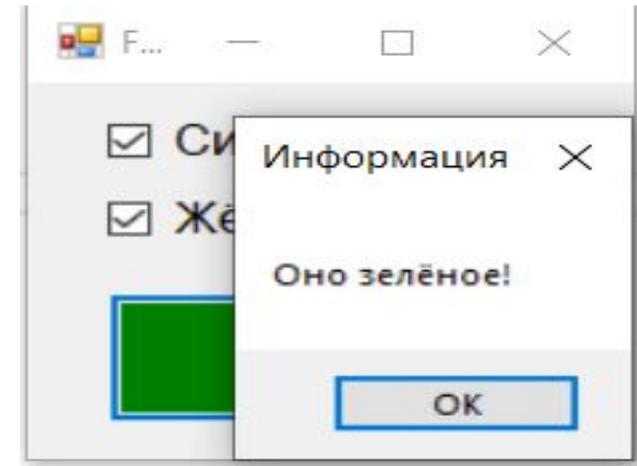
После наведения указателя мыши на кнопку положение данной кнопки на форме должно измениться. Менять положение будем с помощью двух параметров **button2.Left** и **button2.Top**, задающих отступ от границы формы слева и сверху соответственно.

```
private void button2_MouseEnter(object sender, EventArgs e)
{
    Random rnd = new Random();
    button2.Left = rnd.Next(this.Width - 2 * button2.Width);
    button2.Top = rnd.Next(this.Height - 2 * button2.Height);
}
```

MessageBox

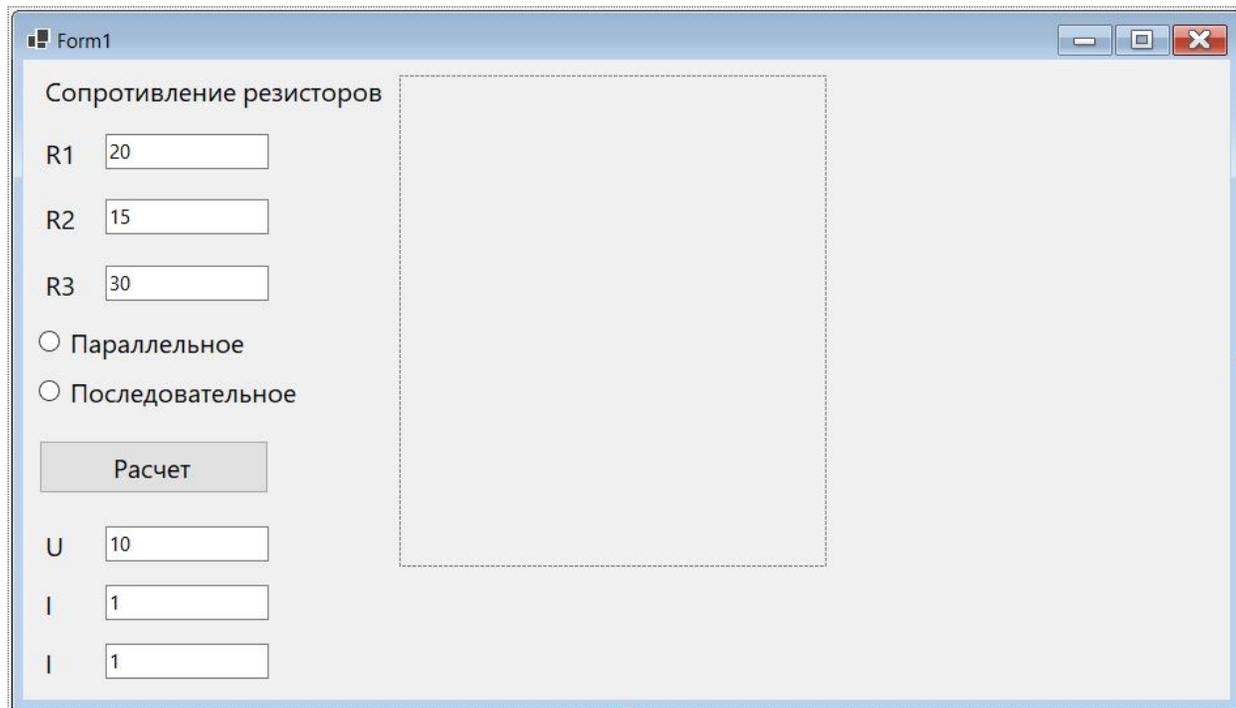
Задание. Для предыдущего примера реализовать вывод пользователю сообщения, информирующего его о цвете кнопки.

```
public void Painting()
{
    button1.BackColor = Color.White;
    if (checkBox1.Checked == true)
        {button1.BackColor = Color.Blue;}
    if (checkBox2.Checked == true)
        {button1.BackColor = Color.Yellow;}
    if ((checkBox1.Checked == true) && (checkBox2.Checked == true))
        {
            button1.BackColor = Color.Green;
            MessageBox.Show("Оно зелёное!", "Информация");
        }
}
```



PictureBox

Задание. Реализовать программу: Проводники с сопротивлением 20 Ом, 15 Ом и 30 Ом соединены последовательно (параллельно). Нарисуйте схему соединения проводников. Определите при заданной силе тока (напряжении) оставшиеся характеристики цепи.



На форме необходимо разместить следующие элементы: PictureBox для отображения схемы соединения проводников, 2 элемента RadioButton для выбора типа соединения (последовательное / параллельное), одну кнопку Button для запуска расчетов, метки и текстовые поля

Для того чтобы в PictureBox отображалась конкретная картинка, нужно прописать путь к ней.

Если расположить картинку в директории проекта (WinFormsApp1\bin\Debug\netcoreapp3.1), то вместо полного пути можно указать только название файла:

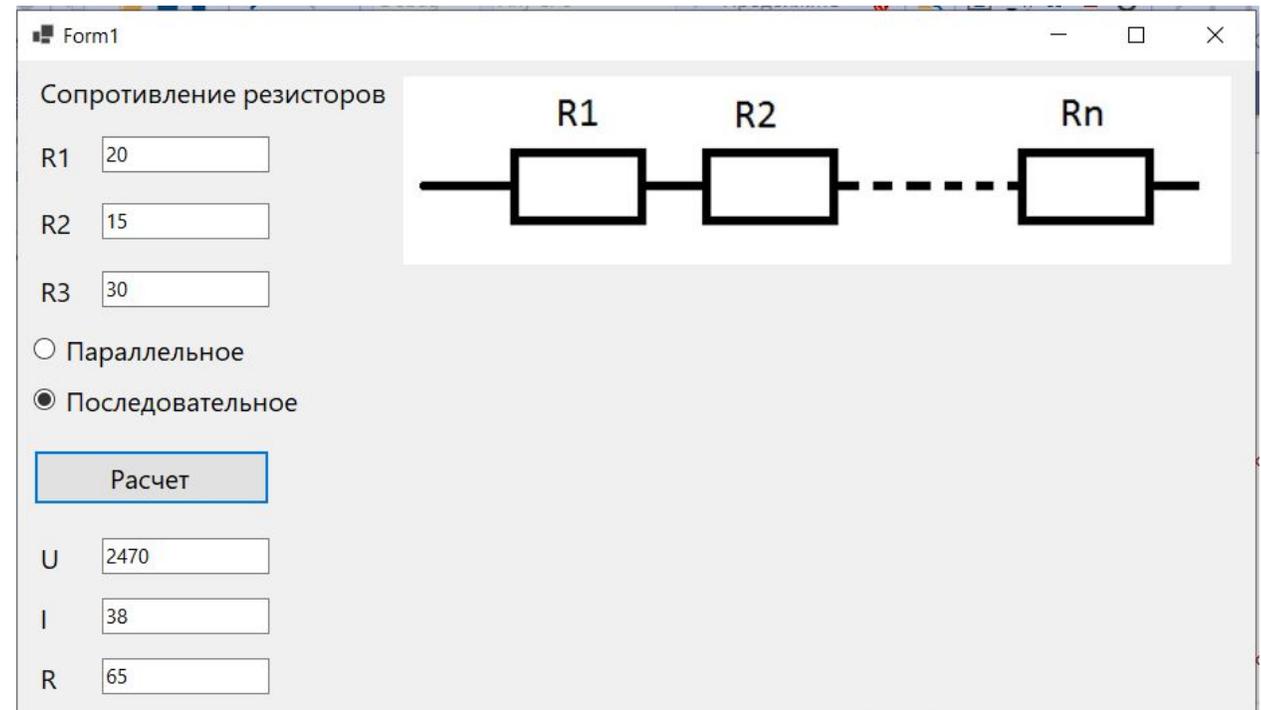
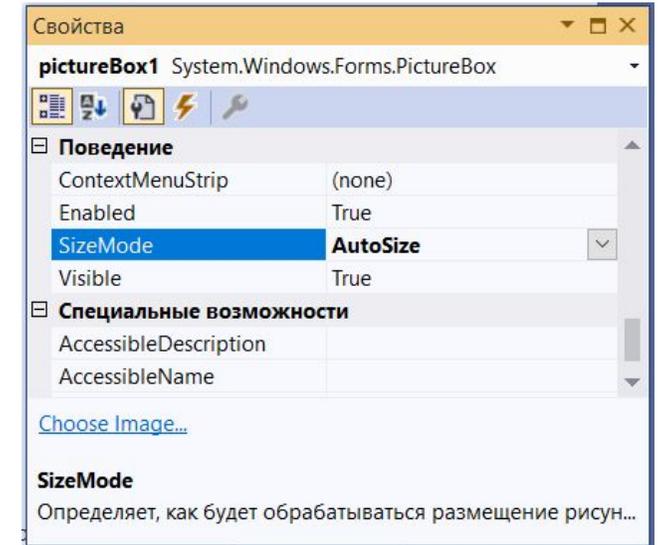
```
pictureBox1.Image = Image.FromFile("screenshot_16.jpg");
```

PictureBox

Для того чтобы вся картинка отображалась в PictureBox, необходимо настроить автоматический размер данного элемента (тогда размер будет подстраиваться под содержимое, т.е. под размер картинки)

Выбор конкретной картинки в соответствии со значением RadioButton:

```
private void button1_Click(object sender, EventArgs e)
{
    if (radioButton1.Checked == true)
    {
        pictureBox1.Image =
Image.FromFile("screenshot_16.jpg");
    }
    if (radioButton2.Checked == true)
    {
        pictureBox1.Image =
Image.FromFile("screenshot_15.jpg");
    }
}
```



Вычисления

я

Реализовать программу для вычисления арифметического выражения при заданных переменных a,b,c.

Исходные данные задачи

a 2

b 14

c 17

Расчет

D

```
private void button1_Click(object sender, EventArgs e)
{
    int a = Convert.ToInt32(textBox1.Text);
    int b = Convert.ToInt32(textBox2.Text);
    int c = Convert.ToInt32(textBox3.Text);
    double d = a / b / c;
    textBox4.Text = Convert.ToString(d);
}
```

Исходные данные задачи

a 2

b 14

c 17

Расчет

D 0

Исходные данные задачи

a 0,25

b 14,76

c 17,42

Расчет

D 0,0009723116748963127

```
private void button1_Click(object sender, EventArgs e)
{
    double a = Convert.ToDouble(textBox1.Text);
    double b = Convert.ToDouble(textBox2.Text);
    double c = Convert.ToDouble(textBox3.Text);
    double d = a / b / c;
    textBox4.Text = Convert.ToString(d);
}
```

Math.Round(d,2)

Исходные данные задачи

a 2

b 14

c 17

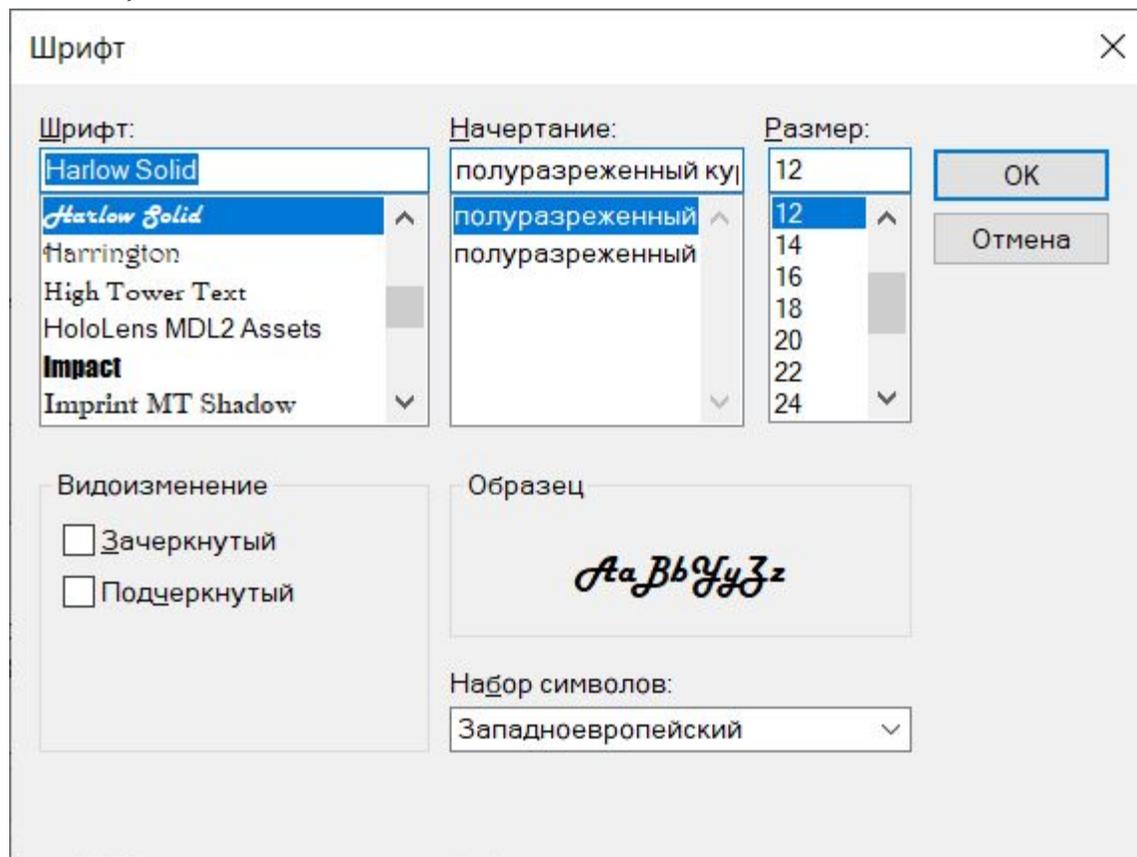
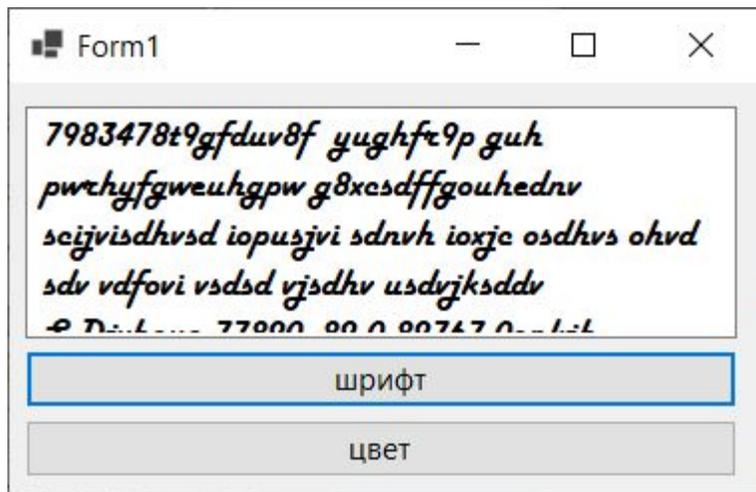
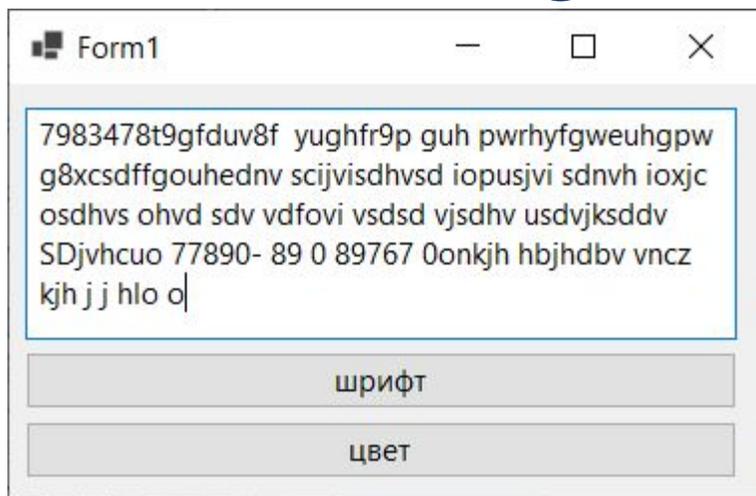
Расчет

D 0,01

fontDialog

Задание. С помощью компонента fontDialog реализовать изменение шрифта в элементе textBox.

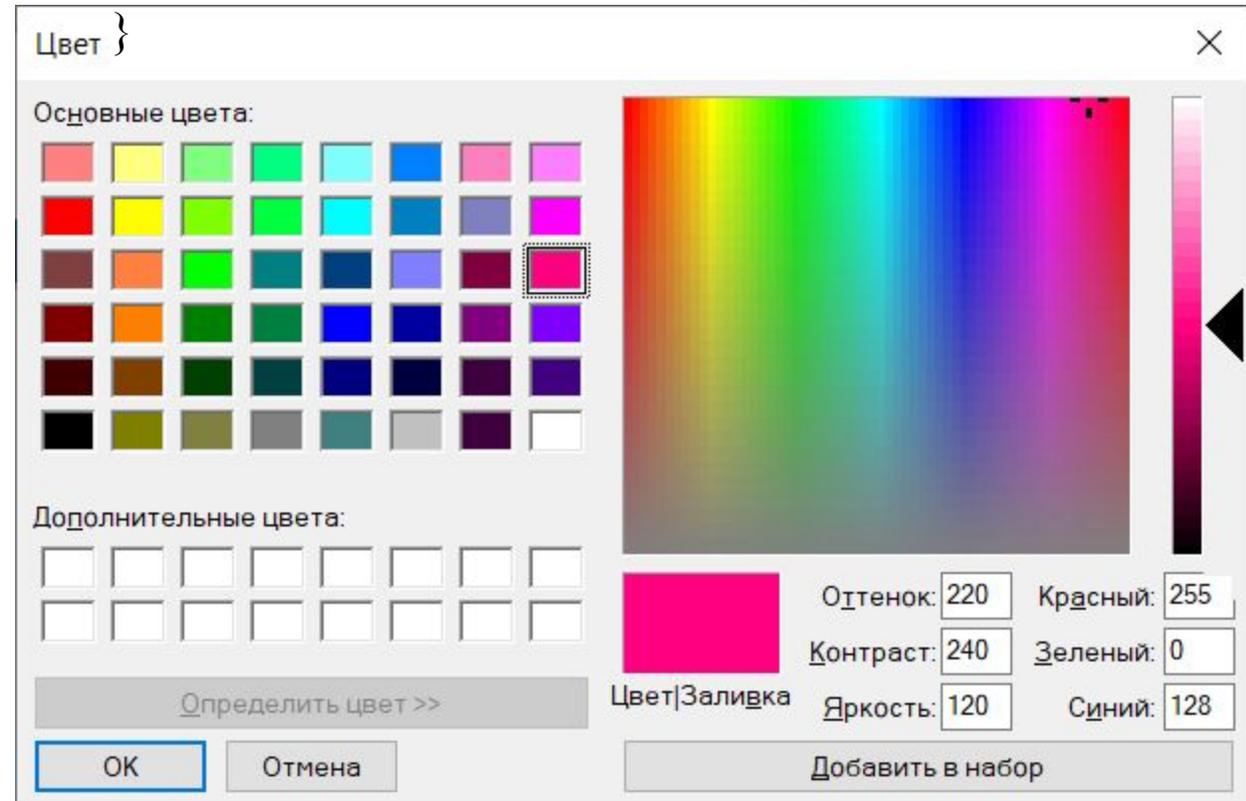
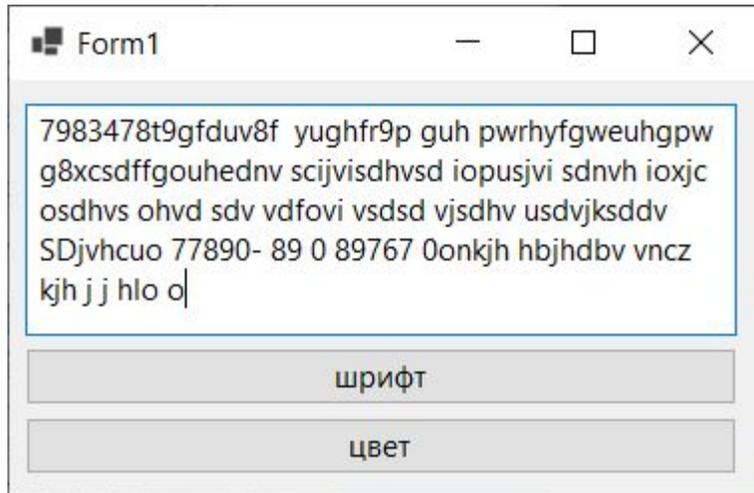
```
private void button1_Click(object sender, EventArgs e)
{
    fontDialog1.ShowDialog();
    textBox1.Font = fontDialog1.Font;
}
```



colorDialog

Задание. С помощью компонента fontDialog реализовать изменение цвета текста в элементе textBox и цвета кнопок.

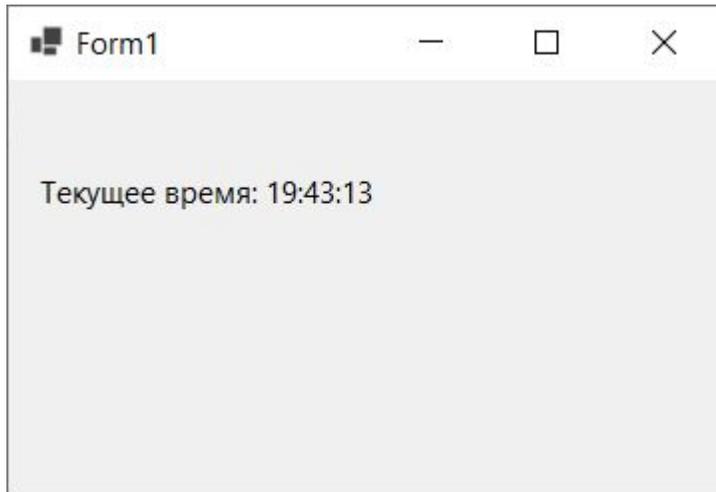
```
private void button2_Click(object sender, EventArgs e)
{
    colorDialog1.FullOpen = true;
    colorDialog1.ShowDialog();
    button1.BackColor = colorDialog1.Color;
    button2.BackColor = colorDialog1.Color;
    textBox1.ForeColor = colorDialog1.Color;
}
```



timer

Задание. С помощью компонента timer реализовать ежесекундный вывод на экран текущего времени.

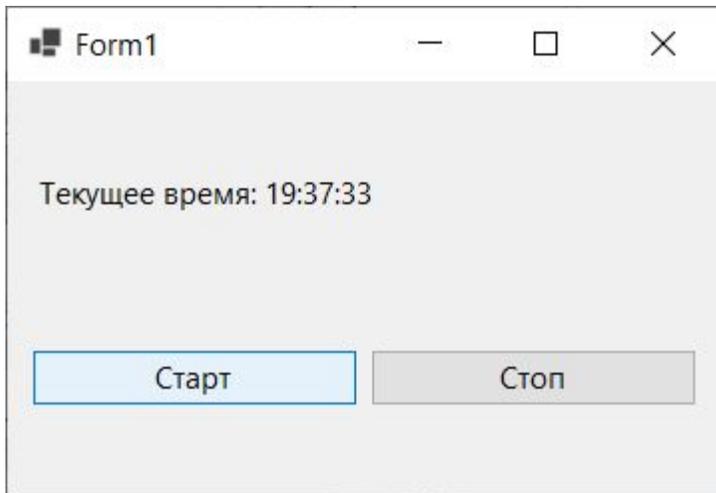
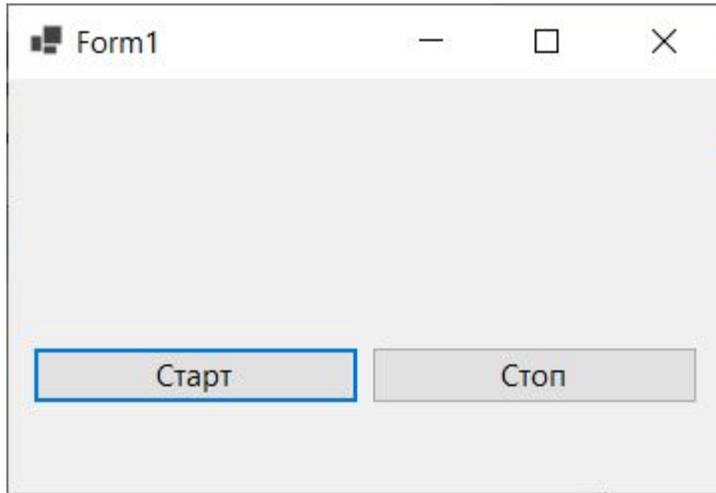
```
private void timer1_Tick(object sender, EventArgs e)
{
    label1.Text = string.Format("Текущее время: {0}", DateTime.Now.ToString("HH:mm:ss"));
}
```



Форматирование – это встраивание в строку различных элементов в заданном формате.

Результат вызова функции `DateTime.Now.ToString("HH:mm:ss")` будет подставлен в строку вместо `{0}`.

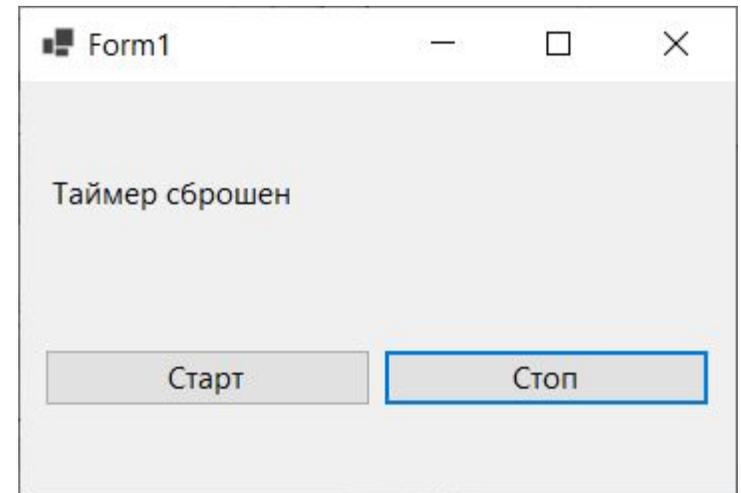
timer



Задание. Для предыдущего задания реализовать запуск и остановку часов.

```
private void button1_Click(object sender, EventArgs e)
{
    timer1.Enabled = true;
    timer1.Start();
}
```

```
private void button2_Click(object sender, EventArgs e)
{
    timer1.Stop();
    timer1.Enabled = false;
    label1.Text = "Таймер сброшен";
}
```



Выводы

- В лекции рассматриваются рабочие области среды разработки Visual Studio 2019, формы, основные элементы (интерфейсные и служебные).
- Приводятся определения понятий «Свойство» и «Событие», объясняются способы взаимодействия с ними через программный код и с помощью редактирования окна свойств.
- Рассматриваются демонстрационные примеры взаимодействия с элементами.



Вопросы по лекции

- Как можно вывести текст в элемент Label?
- Как элемент TextBox сделать недоступным для редактирования?
- Как сделать кнопку на форме невидимой?
- Как называется событие, которое вызывается по нажатию кнопки?
- Чем RadioButton отличается от CheckBox?
- Что представляет собой MessageBox? Какие параметры в него передаются?

