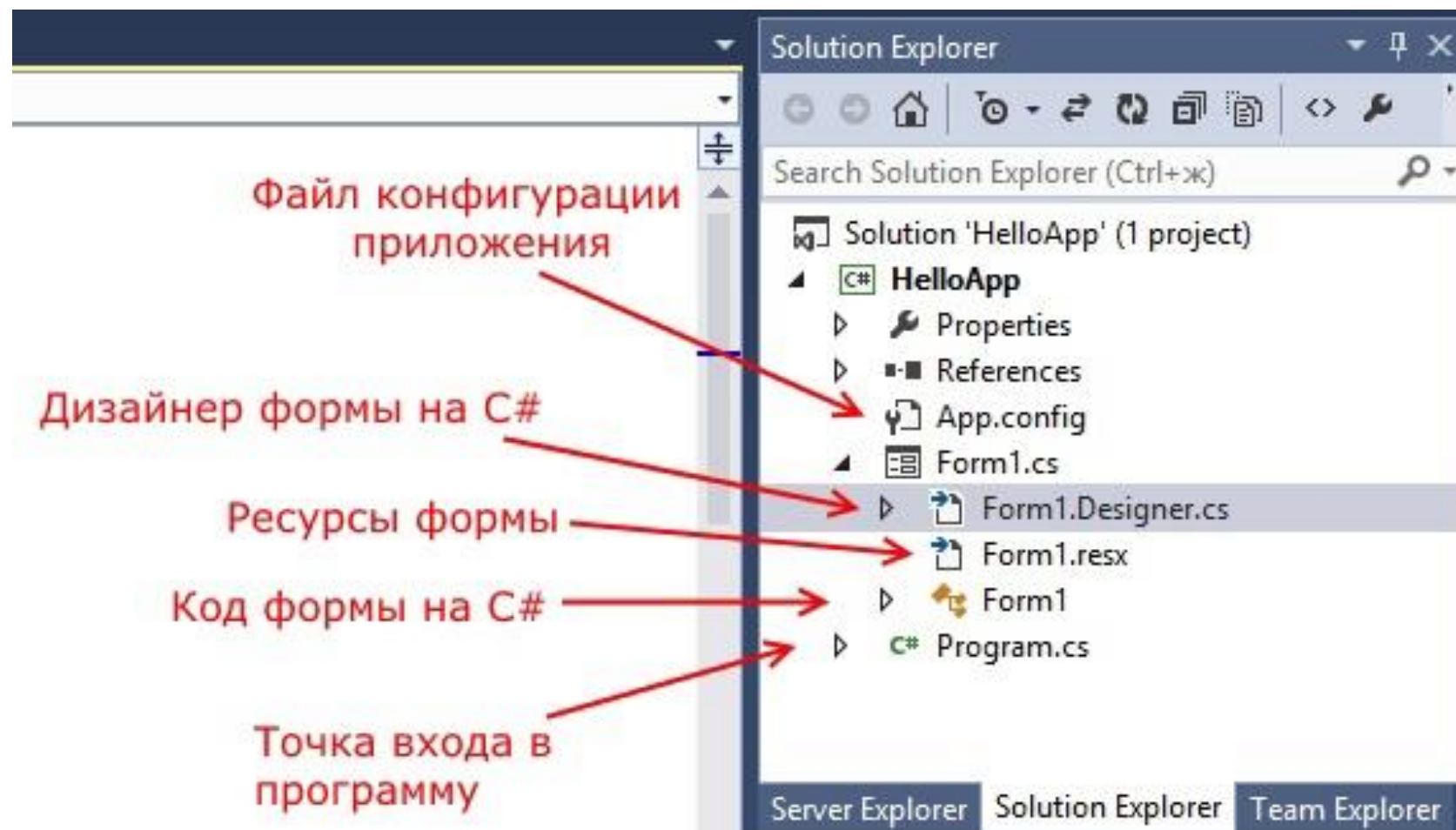


Тема 1: «Создание графического приложения»



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace HelloApp
{
    static class Program
    {
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

```

namespace HelloApp
{
    partial class Form1
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed; otherwise,
false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

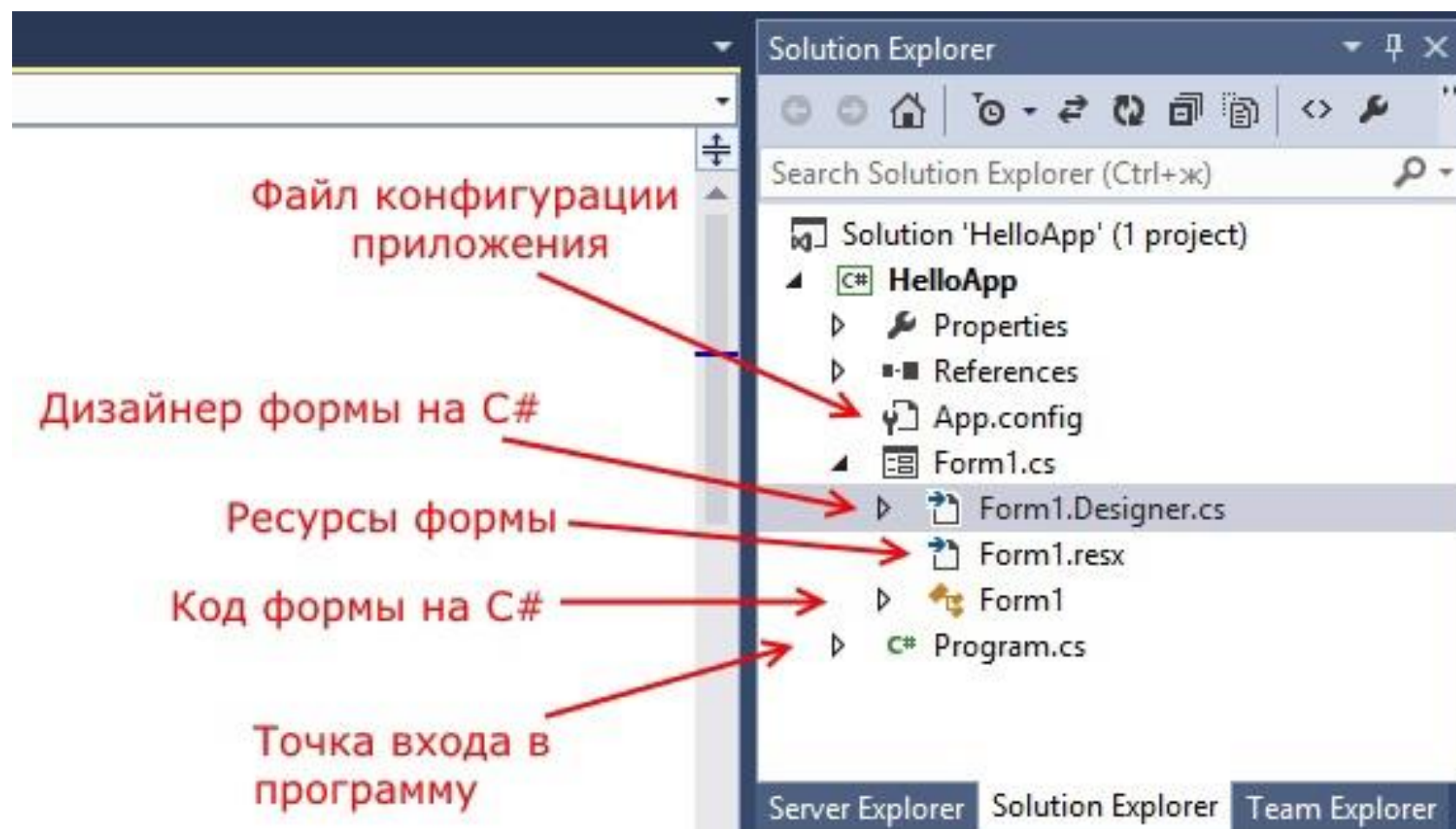
        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            this.SuspendLayout();
            //
            // Form1
            //
            this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
            this.AutoScaleModeMode = System.Windows.Forms.AutoScaleModeMode.Font;
            this.ClientSize = new System.Drawing.Size(284, 261);
            this.Name = "Form1";
            this.Text = "Привет мир!";
            this.ResumeLayout(false);

        }

        #endregion

    }
}

```



```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace HelloApp
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
    }
}

```

```

HelloApp

    class Form1

        <summary>
        ' Required designer variable.
        </summary>
        private System.ComponentModel.IContainer components = null;

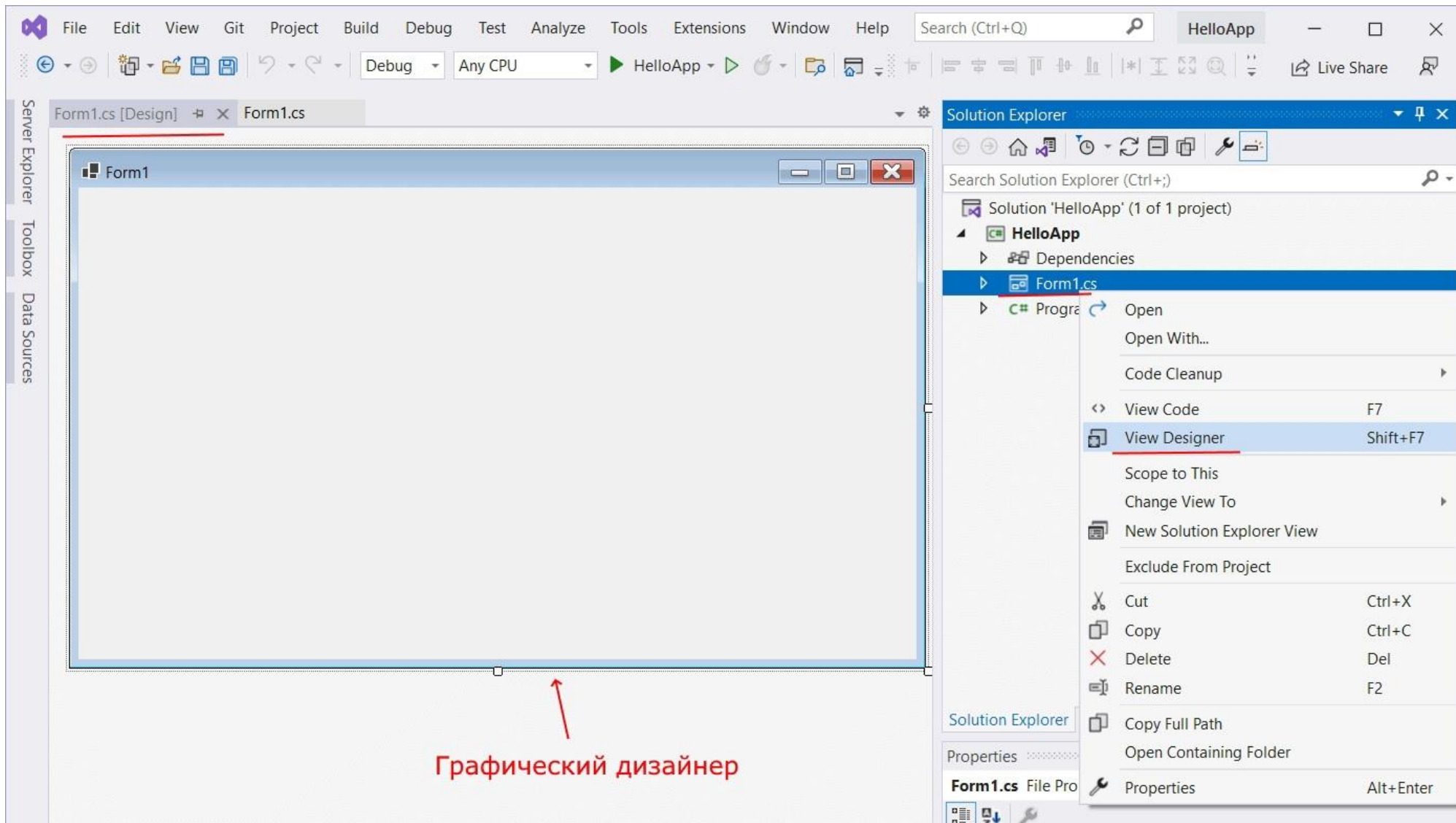
        <summary>
        ' Clean up any resources being used.
        </summary>
        ' <param name="disposing">true if managed resources should be disposed; otherwise,
        ' false; otherwise, if you are using unmanaged resources, you must call Dispose
        ' on them before calling Dispose, using the Dispose(bool) method.
        </param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

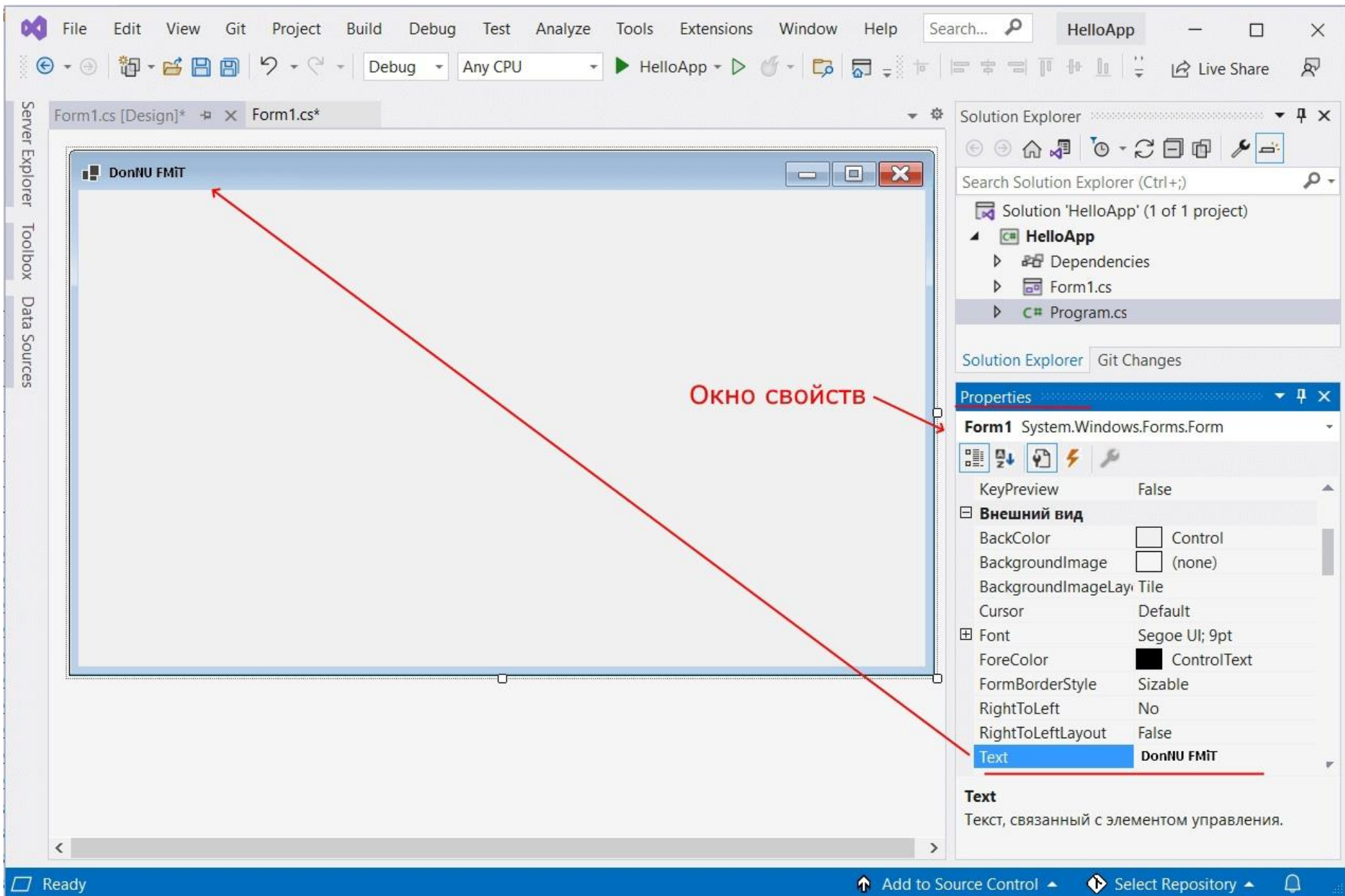
        #region Windows Form Designer generated code

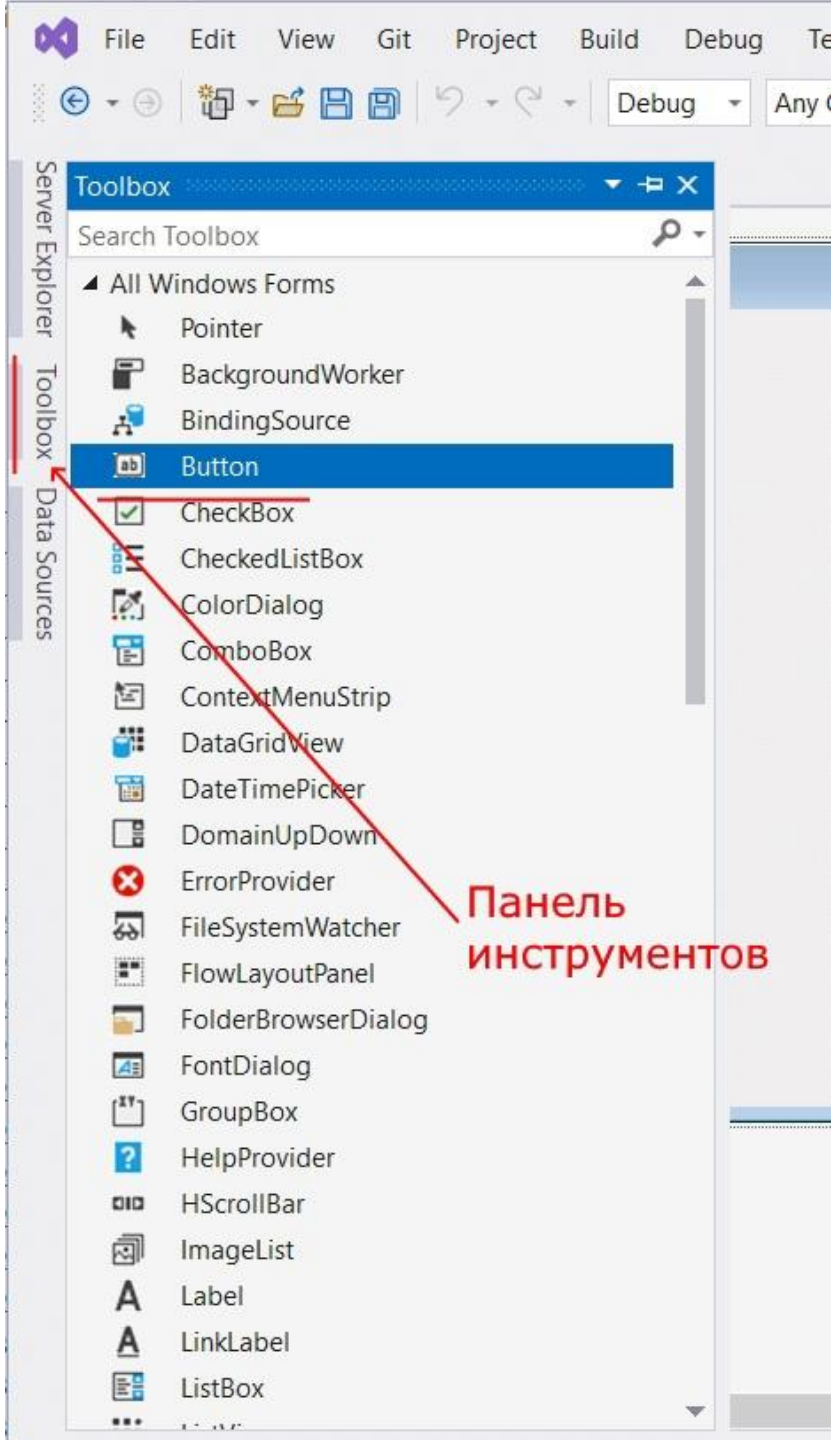
        <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            this.SuspendLayout();
            //
            // Form1
            //
            this.AutoScaleMode = new System.Drawing.SizeF(6F, 13F);
            this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
            this.ClientSize = new System.Drawing.Size(284, 261);
            this.Name = "Form1";
            this.Text = "Привет мир!";
            this.ResumeLayout(false);
        }

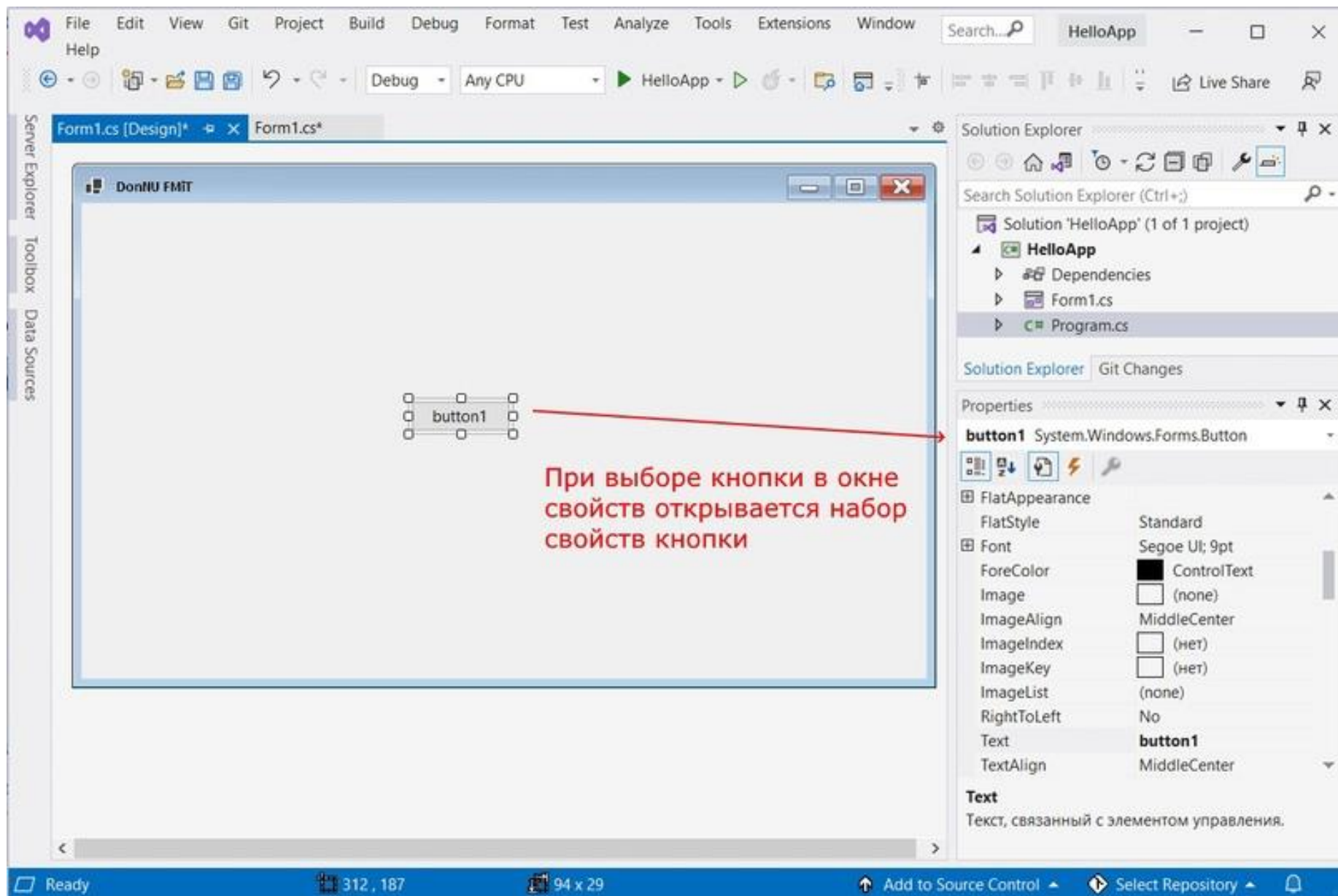
    #endregion

```









```
// button1
//
this.button1.Location = new System.Drawing.Point(312, 187);
this.button1.Name = "button1";
this.button1.Size = new System.Drawing.Size(94, 29);
this.button1.TabIndex = 0;
this.button1.Text = "button1";
this.button1.UseVisualStyleBackColor = true;
//
// Form1
//
this.AutoScaleDimensions = new System.Drawing.SizeF(8F, 20F);
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
this.ClientSize = new System.Drawing.Size(800, 450);
this.Controls.Add(this.button1);
this.Name = "Form1";
this.Text = "DonNU FMiT";
this.ResumeLayout(false);
```

```
namespace HelloApp
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
    }
}
```

```
namespace HelloApp
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();

            button1.Click += button1_Click;
        }

        private void button1_Click(object? sender, EventArgs e)
        {
            MessageBox.Show("Привет");
        }
    }
}
```

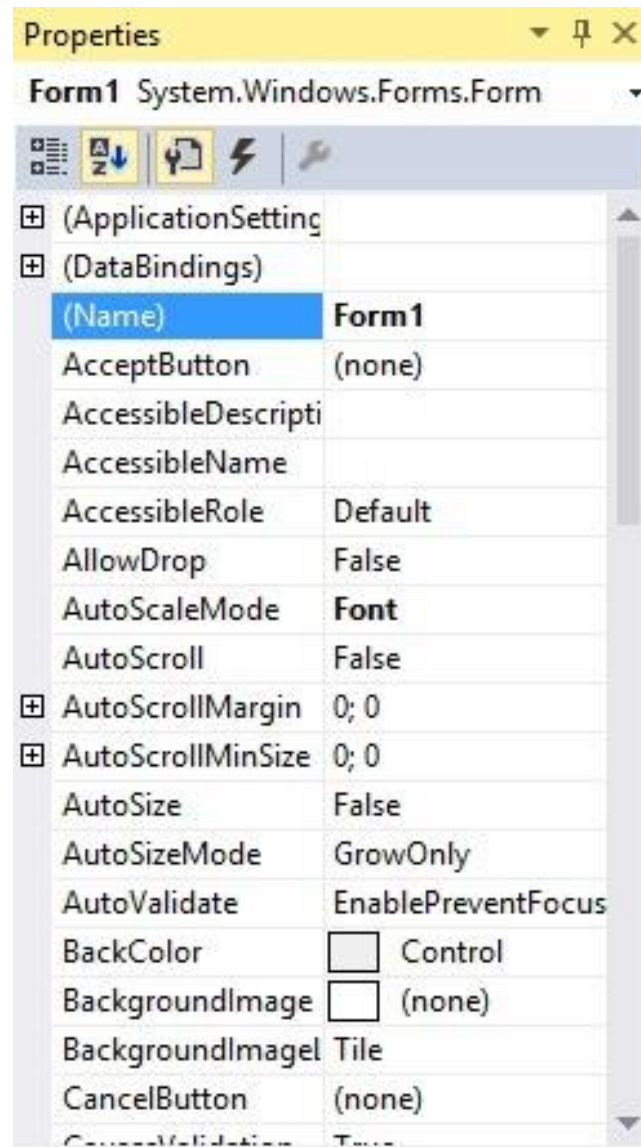
button1



Привет

OK

ОСНОВНЫЕ СВОЙСТВА ФОРМ



(Name)

Indicates the name used in code to identify the object.

Name: устанавливает имя формы - точнее имя класса, который наследуется от класса Form

BackColor: указывает на фоновый цвет формы. Щелкнув на это свойство, мы сможем выбрать тот цвет, который нам подходит из списка предложенных цветов или цветовой палитры

BackgroundImage: указывает на фоновое изображение формы

BackgroundImageLayout: определяет, как изображение, заданное в свойстве BackgroundImage, будет располагаться на форме.

ControlBox: указывает, отображается ли меню формы. В данном случае под меню понимается меню самого верхнего уровня, где находятся иконка приложения, заголовок формы, а также кнопки минимизации формы и крестик. Если данное свойство имеет значение false, то мы не увидим ни иконку, ни крестика, с помощью которого обычно закрывается форма

Cursor: определяет тип курсора, который используется на форме

Enabled: если данное свойство имеет значение false, то она не сможет получать ввод от пользователя, то есть мы не сможем нажать на кнопки, ввести текст в текстовые поля и т.д.

Font: задает шрифт для всей формы и всех помещенных на нее элементов управления. Однако, задав у элементов формы свой шрифт, мы можем тем самым переопределить его

ForeColor: цвет шрифта на форме

FormBorderStyle: указывает, как будет отображаться граница формы и строка заголовка. Устанавливая данное свойство в None можно создавать внешний вид приложения произвольной формы

HelpButton: указывает, отображается ли кнопка справки формы

Icon: задает иконку формы

Location: определяет положение по отношению к верхнему левому углу экрана, если для свойства StartPosition установлено значение Manual

MaximizeBox: указывает, будет ли доступна кнопка максимизации окна в заголовке формы

MinimizeBox: указывает, будет ли доступна кнопка минимизации окна

MaximumSize: задает максимальный размер формы

MinimumSize: задает минимальный размер формы

Opacity: задает прозрачность формы

Size: определяет начальный размер формы

StartPosition: указывает на начальную позицию, с которой форма появляется на экране

Text: определяет заголовок формы

TopMost: если данное свойство имеет значение true, то форма всегда будет находиться поверх других окон

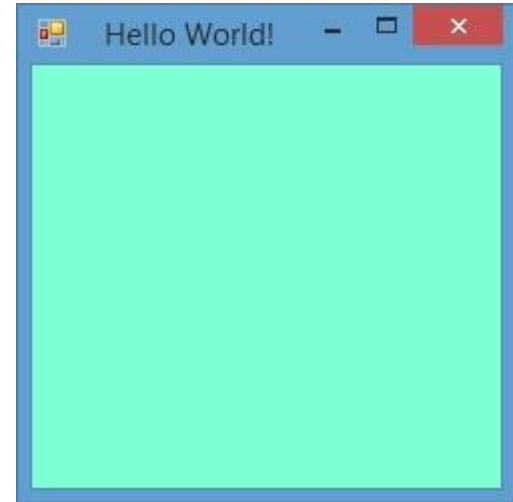
Visible: видима ли форма, если мы хотим скрыть форму от пользователя, то можем задать данному свойству значение false

WindowState: указывает, в каком состоянии форма будет находиться при запуске: в нормальном, maximized или minimized

ПРОГРАММНАЯ НАСТРОЙКА СВОЙСТВ


```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace HelloApp
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
            Text = "Hello World!";
            this.BackColor = Color.Aquamarine;
            this.Width = 250;
            this.Height = 250;
        }
    }
}
```



УСТАНОВКА РАЗМЕРОВ ФОРМЫ

```
this.Size = new Size(200,150);
```

НАЧАЛЬНОЕ РАСПОЛОЖЕНИЕ ФОРМЫ

- **Manual:** Положение формы определяется свойством Location
- **CenterScreen:** Положение формы в центре экрана
- **WindowsDefaultLocation:** Позиция формы на экране задается системой Windows, а размер определяется свойством Size
- **WindowsDefaultBounds:** Начальная позиция и размер формы на экране задается системой Windows
- **CenterParent:** Положение формы устанавливается в центре родительского окна

Все эти значения содержатся в перечислении `FormStartPosition`, поэтому, чтобы, например, установить форму в центре экрана, нам надо прописать так:

```
this.StartPosition = FormStartPosition.CenterScreen;
```

ФОН И ЦВЕТА ФОРМЫ

Чтобы установить цвет как фона формы, так и шрифта, нам надо использовать цветное значение, хранящееся в структуре Color:

```
this.BackColor = Color.Aquamarine;  
this.ForeColor = Color.Red;
```

Кроме того, мы можем в качестве фона задать изображение в свойстве BackgroundImage, выбрав его в окне свойств или в коде, указав путь к изображению:

```
this.BackgroundImage = Image.FromFile("C:\\Users\\Eugene\\Pictures\\3332.jpg");
```

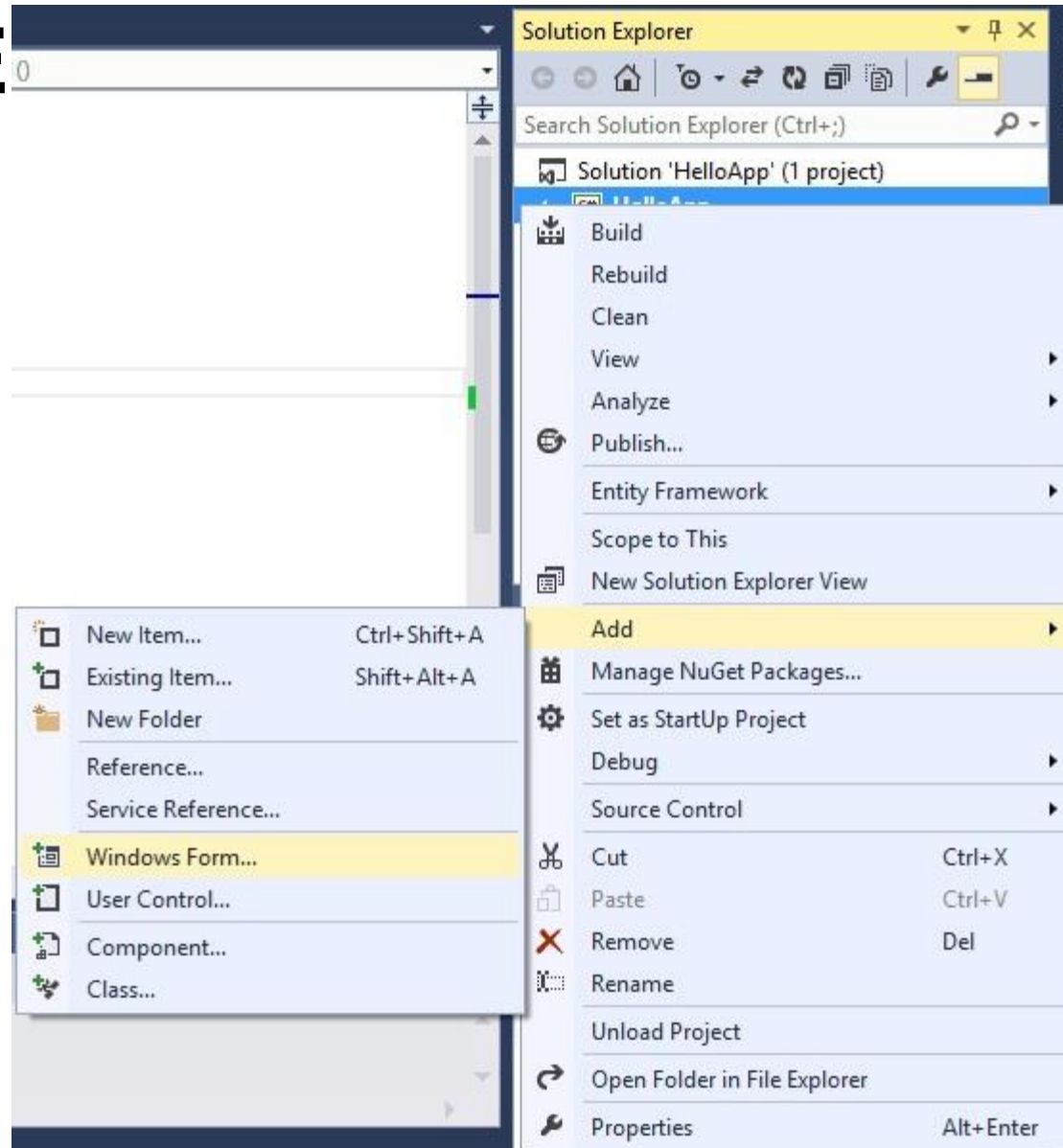
Чтобы должным образом настроить нужное нам отображение фоновой картинки, надо использовать свойство `BackgroundImageLayout`, которое может принимать одно из следующих значений:

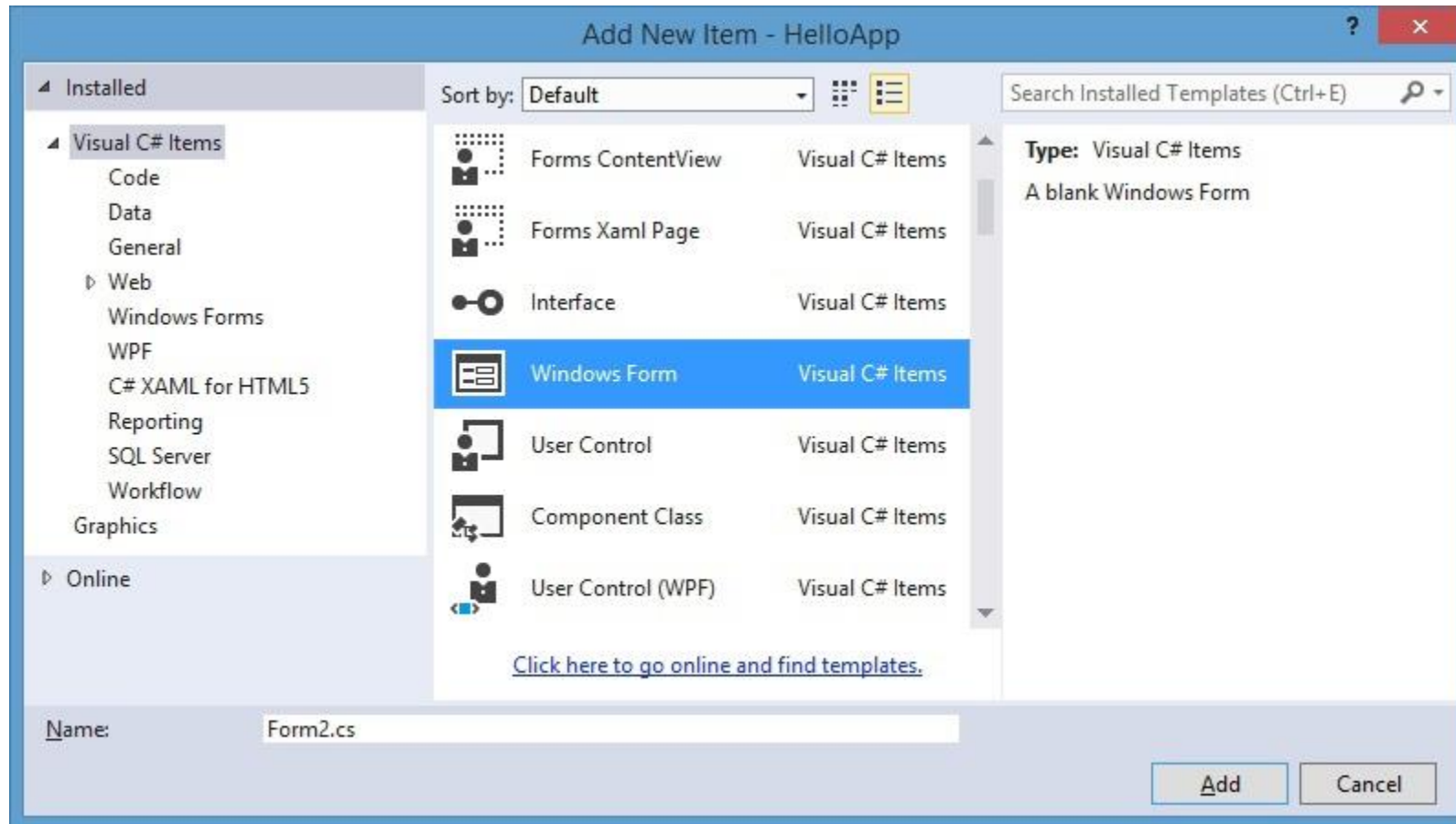
- **None:** Изображение помещается в верхнем левом углу формы и сохраняет свои первоначальные значения
- **Tile:** Изображение располагается на форме в виде мозаики
- **Center:** Изображение располагается по центру формы
- **Stretch:** Изображение растягивается до размеров формы без сохранения пропорций
- **Zoom:** Изображение растягивается до размеров формы с сохранением пропорций

Например, расположим форму по центру экрана:

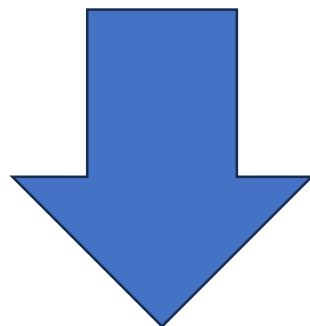
```
this.StartPosition = FormStartPosition.CenterScreen;
```

ДОБАВЛЕНИЕ ФОРМ. ВЗАИМОДЕЙСТВИЕ ФОРМАМИ





```
private void button1_Click(object sender, EventArgs e)
{
}
}
```



```
private void button1_Click(object sender, EventArgs e)
{
    Form2 newForm = new Form2();
    newForm.Show();
}
}
```

```

namespace Names
{
    Ссылка: 3
    public partial class Form1 : Form
    {
        Ссылка: 1
        public Form1()
        {
            InitializeComponent();
        }

        Ссылка: 1
        private void Form1_Load(object sender, EventArgs e)
        {
        }
    }
}

```

```

namespace HelloApp
{
    public partial class Form2 : Form
    {
        public Form2()
        {
            InitializeComponent();
        }

        public Form2(Form1 f)
        {
            InitializeComponent();
            f.BackColor = Color.Yellow;
        }
    }
}

```

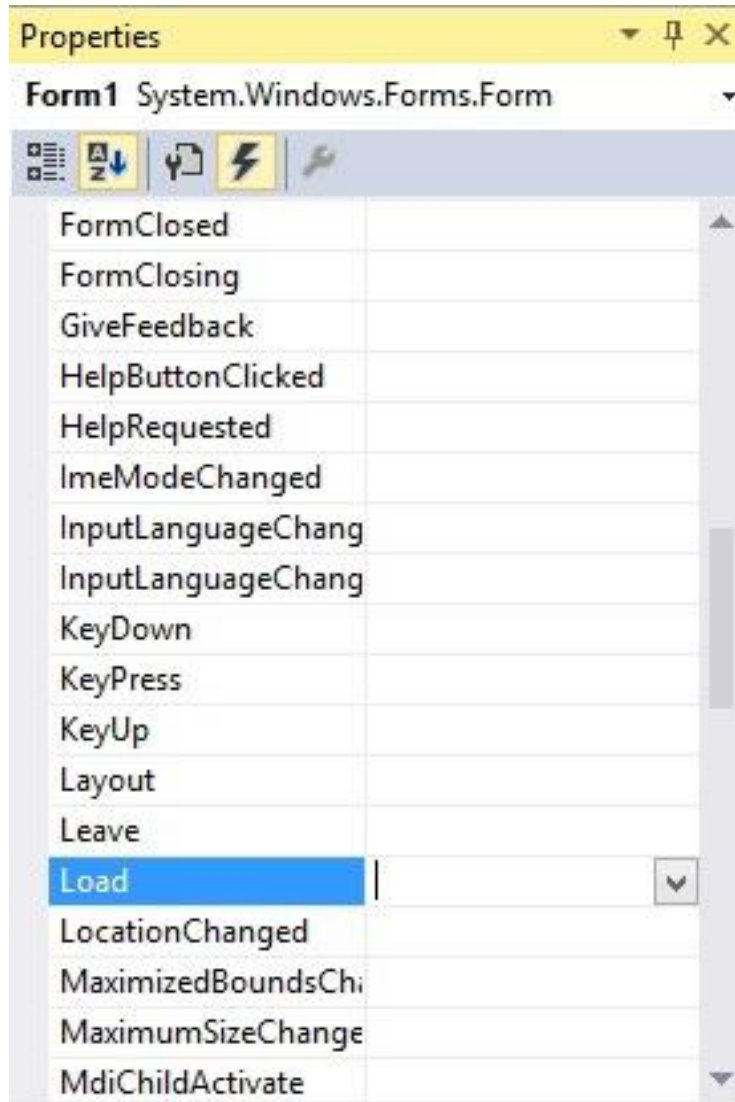
```
private void button1_Click(object sender, EventArgs e)
{
    Form2 newForm = new Form2(this);
    newForm.Show();
}
```

Мы можем также создавать объекты и текущей формы:

```
private void button1_Click(object sender, EventArgs e)
{
    Form1 newForm1 = new Form1();
    newForm1.Show();

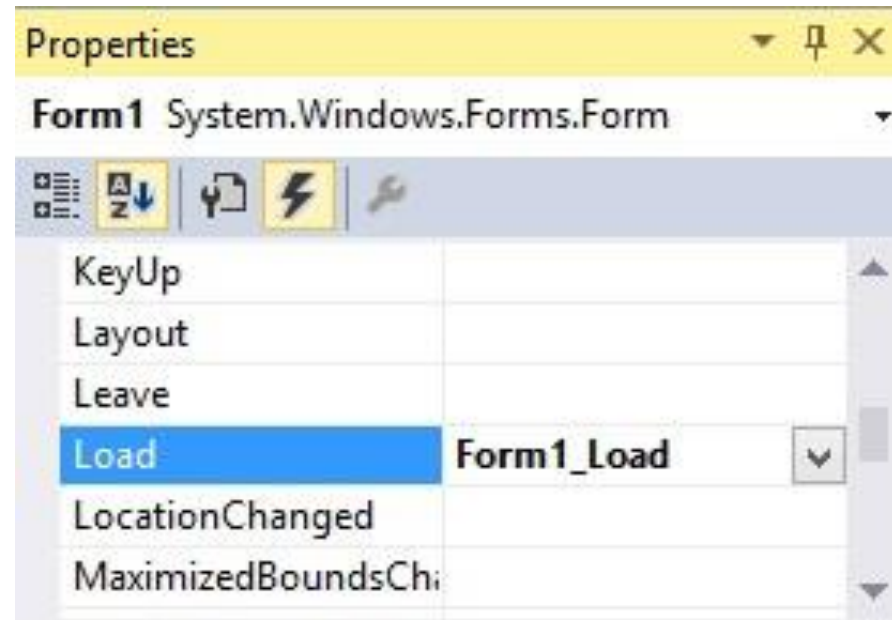
    Form2 newForm2 = new Form2(newForm1);
    newForm2.Show();
}
```

СОБЫТИЯ В WINDOWS FORMS. СОБЫТИЯ ФОРМЫ



Load

Occurs whenever the user loads the form.



```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();
    }

    private void Form1_Load(object sender, EventArgs e)
    {
    }
}
```

Как правило, большинство обработчиков различных визуальных компонентов имеют два параметра: **sender** - объект, инициировавший событие, и аргумент, хранящий информацию о событии (в данном случае EventArgs e).

```
namespace HelloApp
{
    partial class Form1
    {
        private System.ComponentModel.IContainer components = null;

        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }
        private void InitializeComponent()
        {
            this.SuspendLayout();

            this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
            this.AutoScaleMode = System.Windows.Forms.AutoScaleModeMode.Font;
            this.ClientSize = new System.Drawing.Size(284, 261);
            this.Name = "Form1";
            // добавление обработчика
            this.Load += new System.EventHandler(this.Form1_Load);
            this.ResumeLayout(false);
        }
    }
}
```

```
namespace HelloApp
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
            this.Load += LoadEvent;
        }

        private void Form1_Load(object sender, EventArgs e)
        {
        }

        private void LoadEvent(object sender, EventArgs e)
        {
            this.BackColor = Color.Yellow;
        }
    }
}
```

СОЗДАНИЕ НЕПРЯМОУГОЛЬНЫХ ФОРМ. ЗАКРЫТИЕ ФОРМЫ

По умолчанию все формы в Windows Forms являются прямоугольными. Однако мы можем создавать и непрямоугольные произвольные формы. Для этого используется свойство `Region`. В качестве значения оно принимает объект одноименного класса `Region`.

СОЗДАНИЕ НЕПРЯМОУГОЛЬНЫХ ФОРМ. ЗАКРЫТИЕ ФОРМЫ

При создании прямоугольных форм, как правило, не используются границы формы, так как границы задаются этим объектом Region. Чтобы убрать границы формы, надо присвоить у формы свойству `FormBorderStyle` значение `None`.

```

namespace HelloApp
{
    public partial class Form1 : Form
    {
        Point moveStart; // точка для перемещения

        public Form1()
        {
            InitializeComponent();
            this.FormBorderStyle = FormBorderStyle.None;
            this.BackColor = Color.Yellow;
            Button button1 = new Button
            {
                Location = new Point
                {
                    X = this.Width / 3,
                    Y = this.Height / 3
                }
            };
        }
    }
}

```

```

button1.Text = "Заккрыть";
button1.Click += button1_Click;
this.Controls.Add(button1); // добавляем кнопку на форму
this.Load += Form1_Load;
this.MouseDown += Form1_MouseDown;
this.MouseMove += Form1_MouseMove;

```

```

private void button1_Click(object sender, EventArgs e)
{
    this.Close();
}

```

```

private void Form1_Load(object sender, EventArgs e)
{
    System.Drawing.Drawing2D.GraphicsPath myPath = new
System.Drawing.Drawing2D.GraphicsPath();
    // создаем эллипс с высотой и шириной формы
    myPath.AddEllipse(0, 0, this.Width, this.Height);
    // создаем с помощью эллипса ту область формы, которую мы хотим видеть
    Region myRegion = new Region(myPath);
    // устанавливаем видимую область
    this.Region = myRegion;
}

```

```

private void Form1_MouseDown(object sender, MouseEventArgs e)
{
    // если нажата левая кнопка мыши
    if (e.Button == MouseButtons.Left)
    {
        moveStart = new Point(e.X, e.Y);
    }
}

```

```

private void Form1_MouseMove(object sender, MouseEventArgs e)
{
    // если нажата левая кнопка мыши
    if ((e.Button & MouseButtons.Left) != 0)
    {
        // получаем новую точку положения формы
        Point deltaPos = new Point(e.X - moveStart.X, e.Y - moveStart.Y);
        // устанавливаем положение формы
        this.Location = new Point(this.Location.X + deltaPos.X,
            this.Location.Y + deltaPos.Y);
    }
}
}
}

```


Point mo

p

Закреть

}

private void {object sender