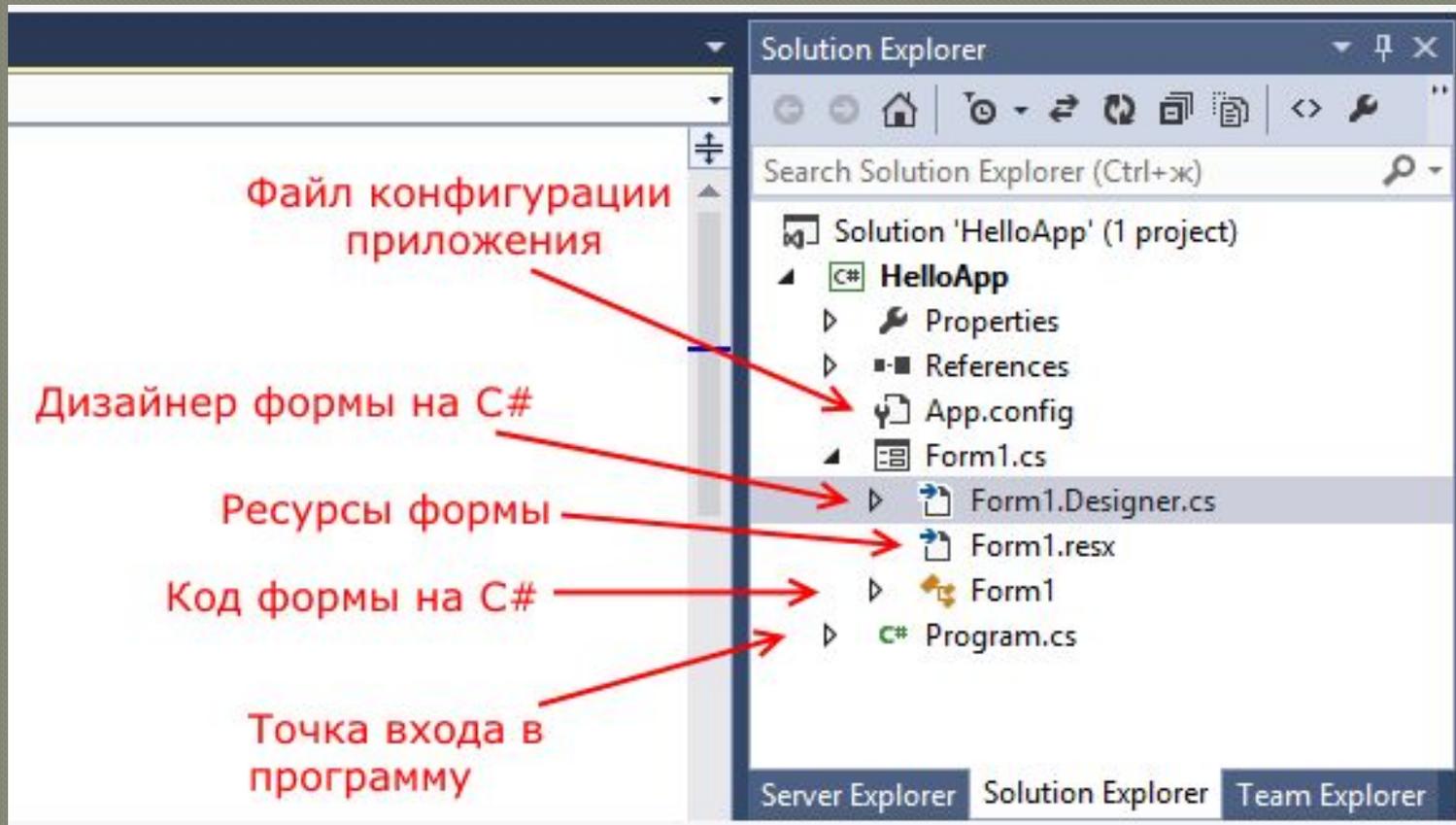


# Работа с формами

# Компоненты формы



# Свойства, оказывающие влияние на визуальное отображение формы.

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

# ВЛИЯНИЕ НА ВИЗУАЛЬНОЕ ОТОБРАЖЕНИЕ ФОРМЫ.

- **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
- **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
- **Например:** `this.BackColor = Color.Aquamarine;`

# формы

- Начальное расположение формы устанавливается с помощью свойства `StartPosition`, которое может принимать одно из следующих значений:
- **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
- **Например:**
- `this.StartPosition = FormStartPosition.CenterScreen;`

# Задание фона:

---

- в качестве фона можно 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");
```

# свойство BackgroundImageLayout

- Позволяет настроить нужное отображение фоновой картинки, 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 формы в проект:

- Чтобы добавить еще одну форму в проект, нажмем на имя проекта в окне Solution Explorer (Обозреватель решений) правой кнопкой мыши и выберем Add(Добавить)->Windows Form...

## Взаимодействие между двумя формами.

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

# Динамическое добавление элементов

---

- добавим событие загрузки формы, в котором будет создаваться новый элемент управления.
- `Button helloButton = new Button();`/код добавления кнопки на форму
- `helloButton.Location = new Point(10, 10);`/положение кнопки
- `this.Controls.Add(helloButton);`/добавление кнопки в коллекцию элементов формы

# Кнопка

---

Чтобы управлять внешним отображением кнопки, можно использовать свойство **FlatStyle**.

Оно может принимать следующие значения:

- **Flat** - Кнопка имеет плоский вид
- **PopUp** - Кнопка приобретает объемный вид при наведении на нее указателя, в иных случаях она имеет плоский вид
- **Standard** - Кнопка имеет объемный вид (используется по умолчанию)
- **System** - Вид кнопки зависит от операционной системы

# Изображение на кнопке

---

- Для этого надо использовать свойство **TextImageRelation**. Оно приобретает следующие значения:
- **Overlay**: текст накладывается на изображение
- **ImageAboveText**: изображение располагается над текстом
- **TextAboveImage**: текст располагается над изображением
- **ImageBeforeText**: изображение располагается перед текстом
- **TextBeforeImage**: текст располагается перед изображением

# Автозаполнение текстового поля

---

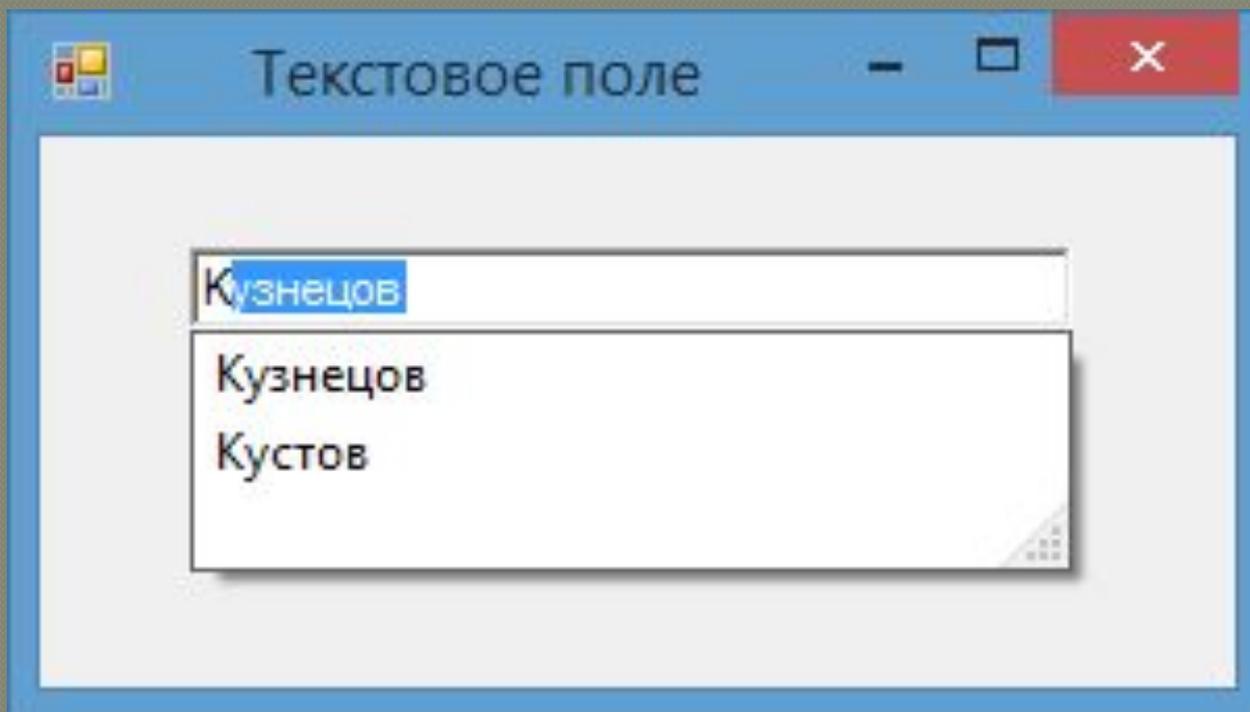
- Элемент `TextBox` обладает достаточными возможностями для создания автозаполняемого поля.
- Для этого нам надо привязать свойство `AutoCompleteCustomSource` элемента `TextBox` к некоторой коллекции, из которой берутся данные для заполнения поля.

---

```
public partial class Form1 : Form
{
    public Form1 ()
    {
        InitializeComponent();
        AutoCompleteStringCollection source = new
AutoCompleteStringCollection()
        {
            "Кузнецов",
            "Иванов",
            "Петров",
            "Кустов"
        };
        textBox1.AutoCompleteCustomSource = source;
        textBox1.AutoCompleteMode =
AutoCompleteMode.SuggestAppend;
        textBox1.AutoCompleteSource =
AutoCompleteSource.CustomSource;
```

# Результат работы

---



# Задание.

---

- Создать проект из 5 форм, обеспечить возможность перехода с формы на форму;
- Изменить курсор мыши, задать для первых трех форм определенный цвет
- На каждую кнопку добавить изображение, сделать кнопки объемными и плоскими

- 
- **На первой форме изменить цвета текста в форме**
  - **Вторую форму сделать прозрачной**
  - **Использовать в качестве фона второй формы картинку**
  - **Задать положение четвертой формы в центре экрана**
  - **На последней форме сделать автозаполнение текстового поля**
  - ***Дополнительное задание: сделать последнюю форму круглой формы***