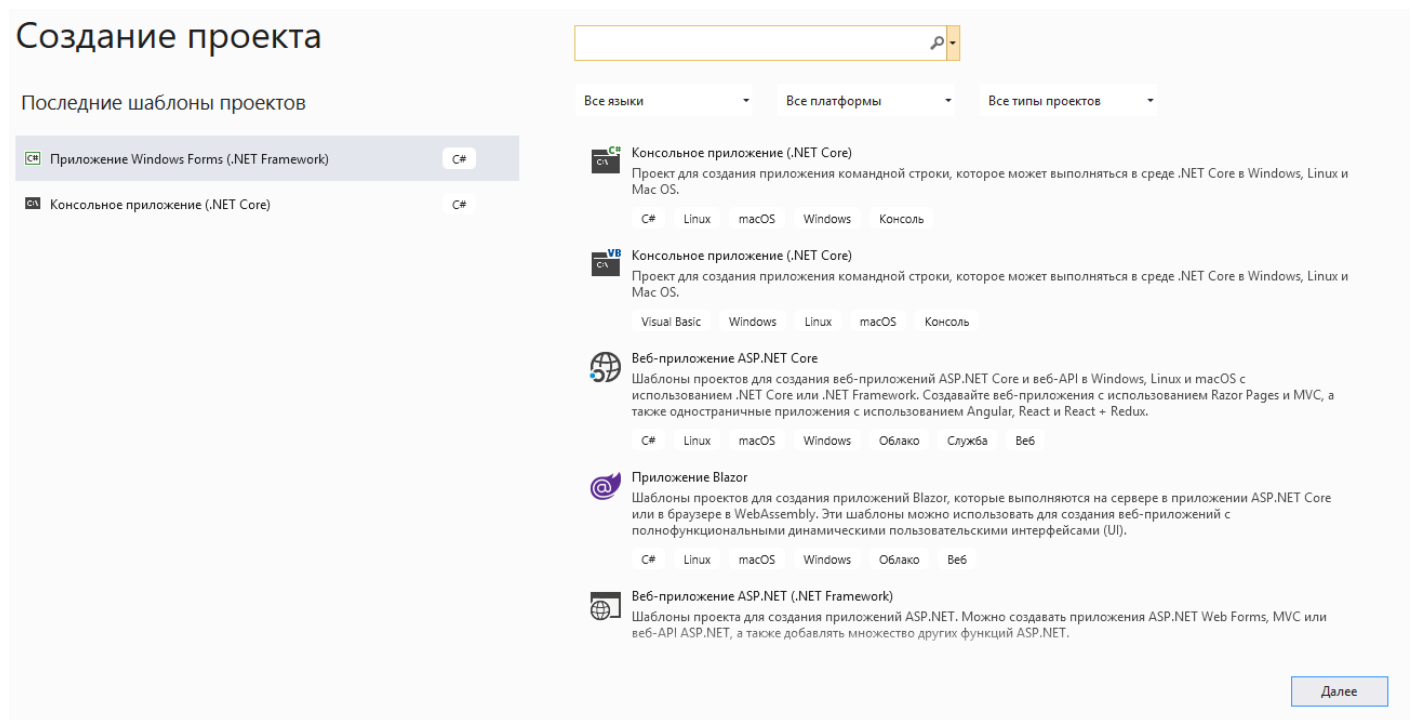


Задание 1. Разработка формы. Button, Label, TextBox, PictureBox, элементы выбора

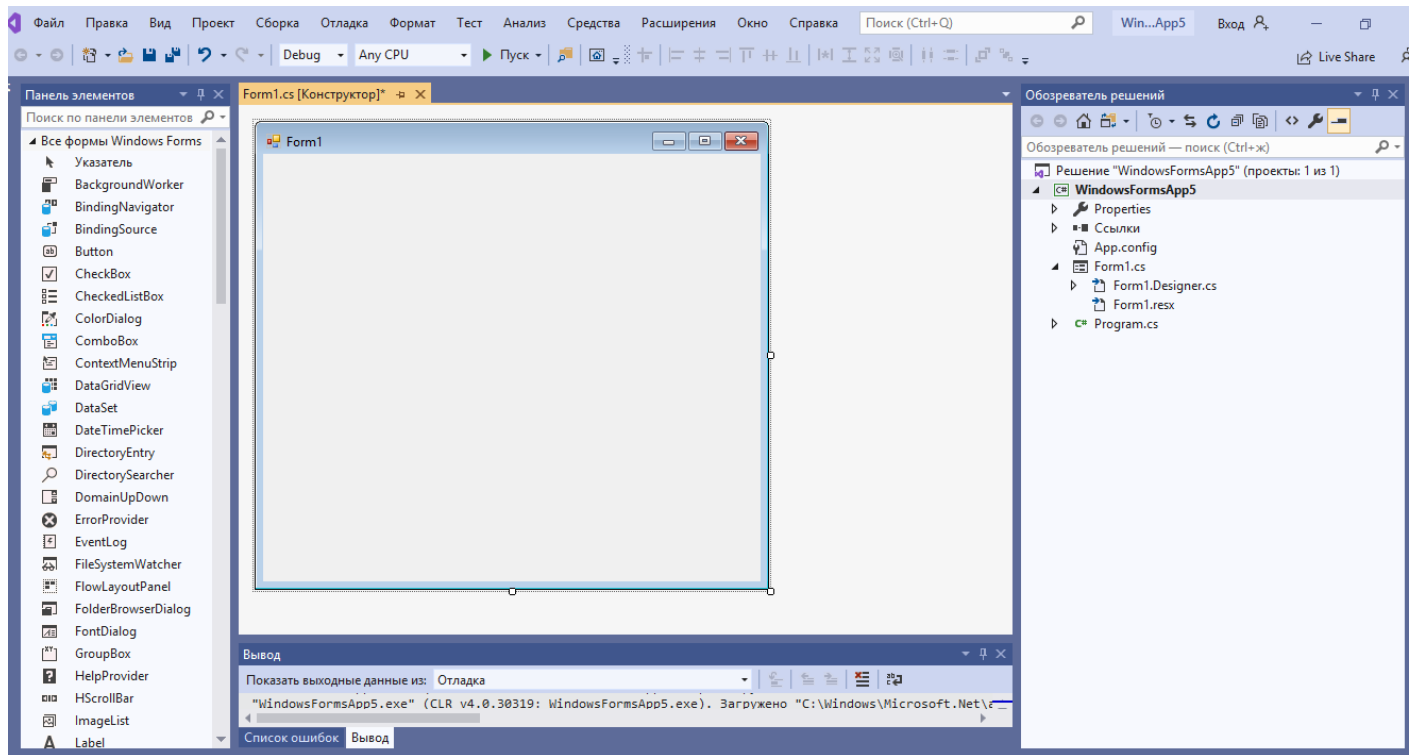
Задача 1.

Создайте новый проект с помощью команды File/ New Project или Create Project при открытии Visual Studio. На экране появляется диалоговое окно с шаблонами проектов



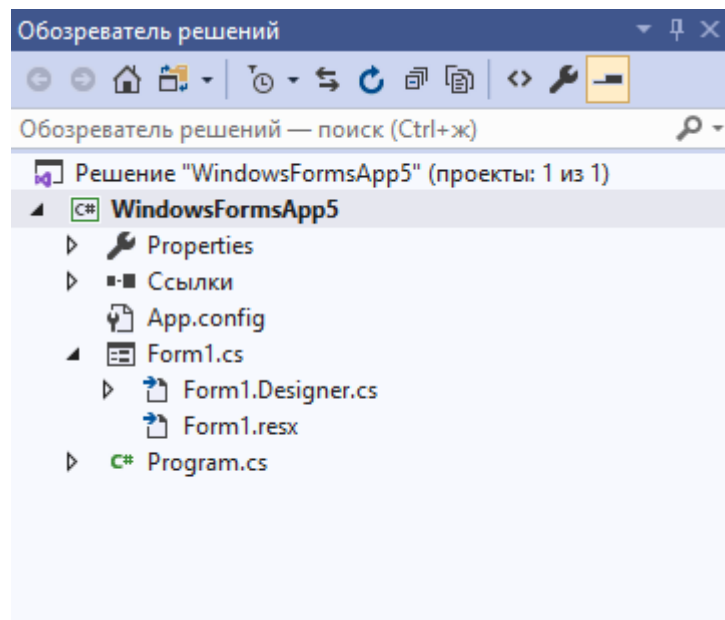
Выберите шаблон проекта Windows Forms (.Net Framework). Дайте имя проекту, например, «Задача_1».

После нажатия на кнопку Ок, откроется окно интегрированной среды разработки Visual Studio **IDE** (Integrated Development Environment).



Мастер создания нового проекта создает новый класс Form1, производный от System.Windows.Forms.Form.

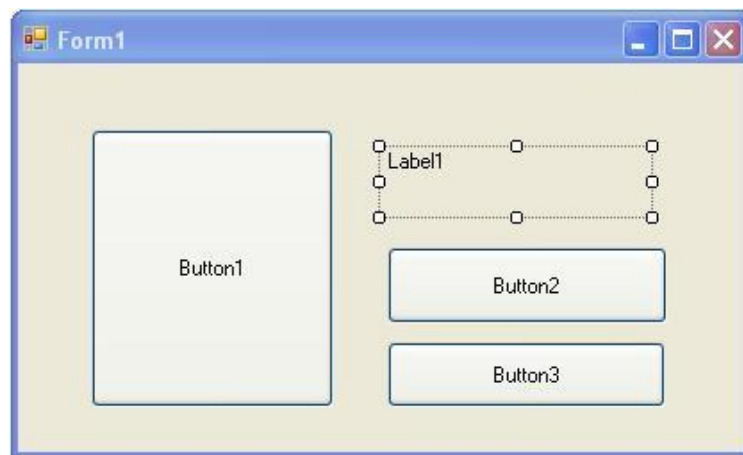
В окне решения (Solution Explorer) выводится список файлов, входящих в решение. По умолчанию имя решения совпадает с именем первого, созданного в нем проекта.



В C# всем файлам присваивается расширение .cs. По умолчанию файл с главной формой приложения называется Form1.cs.

Панель элементов (Toolbox) используется для добавления в приложение элементов управления. При этом, IDE автоматически будет генерировать код для этих элементов.

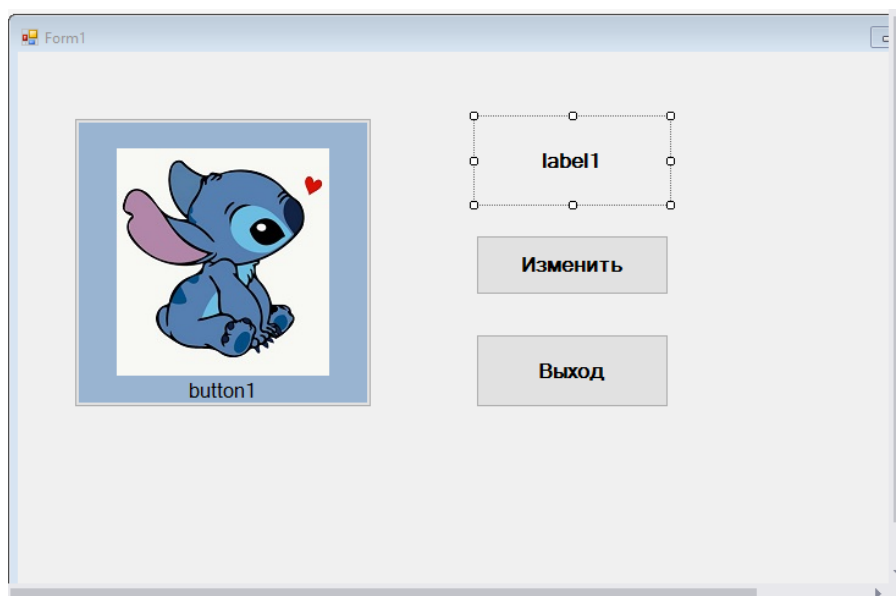
С панели элементов ToolBox перетаскивайте мышью на форму следующие элементы управления.



Чтобы увеличить размер метки, установите в окне свойств Properties свойство `AutoSize` равное `False`.

Для задания графического изображения, помещаемого на кнопке, выделите свойство `Image` и нажмите кнопку, расположенную в правом столбце. В диалоговом окне выберите на диске файл, содержащий изображение. Изображение должно быть небольшим – по размеру кнопки.

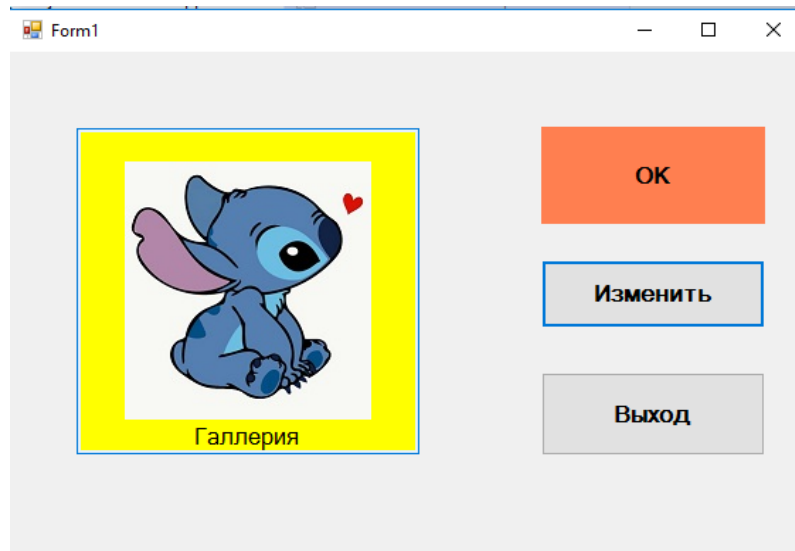
Измените свойства `Text`, `Font`, `BackColor`, `TextAlign` формы и элементов управления следующим образом:



Цвет формы задайте любым понравившимся вам цветом. Задайте нестандартный цвет фона `Button1`. Для этого в свойстве `BackColor` на вкладке `Custom` на нижних пустых клеточках откройте с помощью контекстного меню (правая кнопка мыши) `Окно формирования цвета`. Подберите нужный цвет и нажмите кнопку `Add` (добавить).

Для **Label1** в свойстве `TextAlign` установите выравнивание по центру.

При нажатии на кнопку **Изменить** форма должна приобрести следующий вид:



С помощью программного кода меняются значения Text и BackColor для метки Label1 и кнопки Button1. Введите следующую процедуру в окне редактора кода (двойной щелчок мыши по кнопке Button2, чтобы вызвать редактор кода):

```
private void button2_Click(object sender, EventArgs e)
{
    button1.BackColor=Color.Yellow;
    button1.Text="Галерея";
    label1.BackColor = Color.Coral;
    label1.Text = "Ok";
}
```

Цвет фона Button1 определяется с помощью константы структуры Color: Color.Yellow, значение которой равно коду желтого цвета. Другие цвета можно выбрать из списка, открывающегося для структуры Color.

При нажатии на кнопку Галерея должна открыться новая форма с изображением.

Для этого добавьте в проект еще одну форму с помощью команды Project/Add Windows Form или с помощью пиктограммы Add New Item.

На форме поместите элемент управления графическое окно **PictureBox** и командную кнопку **Button** для выхода из формы.

Для **PictureBox** в свойстве Image выберите файл, содержащий изображение.

На командную кнопку **Button** поместите изображение.



В процедуре нажатия кнопки введите оператор, закрывающий форму:

```
this.Visible = false;
```

this – имя текущего (активного) объекта. В данном случае закрывается текущая форма.

Visible – свойство, указывающее на видимость объекта. Так как оно относится к текущему объекту, то можно использовать его без ключевого слова this:

```
Visible = false;
```

Кнопка button1 формы Form1 должна открывать форму Form2. Для этого в процедуре button1_Click нужно ввести следующий код:

```
Form2 frm2 = new Form2();  
frm2.Show();
```

В первой строчке создаётся объект второй формы, а во второй - метод её вызова.

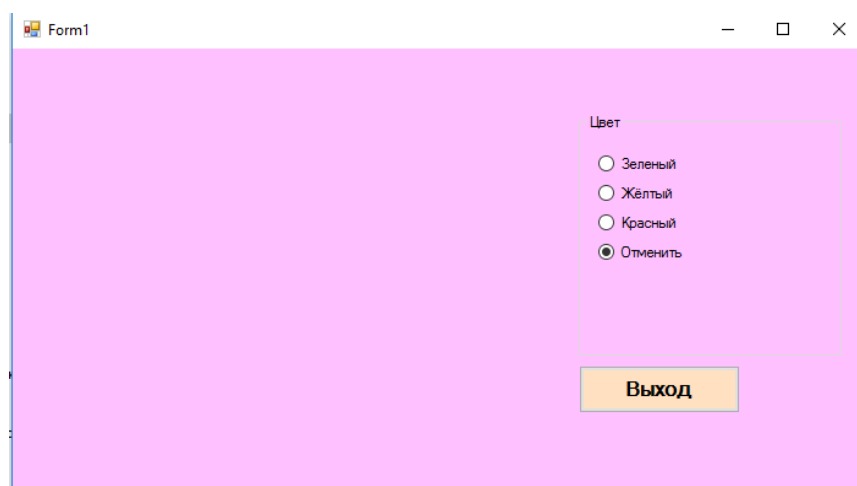
Для кнопки Button3 формы Form1 создайте процедуру с оператором, закрывающим форму Form2:

```
this.Close();
```

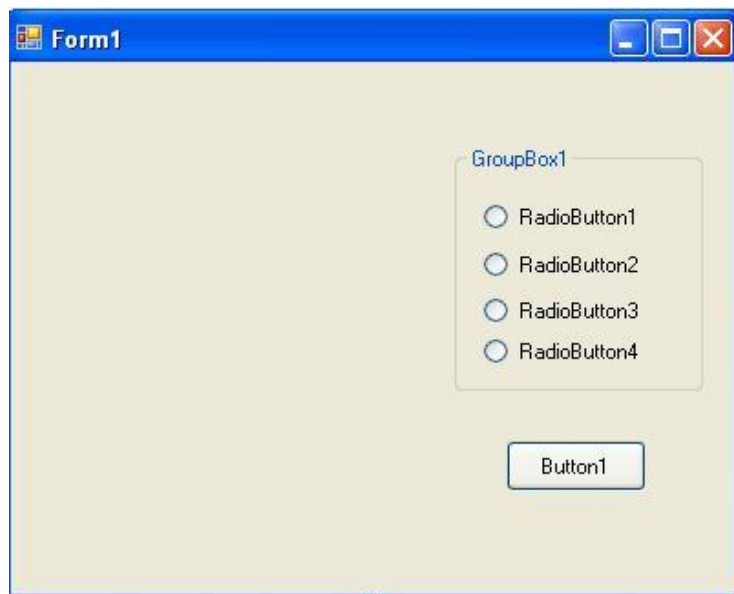
Или просто

```
Close();
```

Задача 2. Необходимо создать форму для изменения цвета формы. Таким образом должна выглядеть форма для решения этой задачи.



1. Добавьте на форму следующие элементы управления:



2. В этой форме для выбора цвета используется элемент управления переключатели (RadioButton). Это набор элементов для выбора одного из многих вариантов. Кнопки переключатели обязательно объединяются в группы (не менее двух в группы). Объединяются они с помощью элемента GroupBox. Сначала на форму поместите GroupBox, а затем внутри разместите переключатели.

GroupBox – это объект-контейнер. При перемещении вместе с ней перемещаются все элементы, помещенные в рамке. Уничтожая GroupBox, уничтожаются все элементы внутри.

Если нажата одна кнопка, то у нее свойство `Checked = True`, у остальных – `False`.

3. Измените свойства `Text` для всех элементов формы соответственно первому рисунку.

4. Задайте нестандартный цвет фона формы (свойство `BackColor`).

5. Для переключателя `RadioButton1` для события `CheckedChanged` создайте процедуру с одним оператором, изменяющим цвет формы на зеленый

6. Для двух других переключателей задайте соответствующий цвет

Для переключателя Отмена необходимо восстановить первоначальный цвет формы.

Для задания первоначального цвета можно использовать метод `FromArgb` структуры `Color`. Этот метод задает цвет с помощью палитры ARGB (значения `Red`, `Green`, `Blue`):

```
BackColor = Color.FromArgb(192, 225, 226);
```

Значения параметров (`Red`, `Green`, `Blue`) задавались в окне свойств при выборе цвета. Оттуда эти значения можно скопировать.

Убедитесь, что для **RadioButton** в свойстве **Checked** выставлено значение **False** (даже если оно стоит там по умолчанию). Это гарантирует, что при запуске ни один переключатель не будет выбран.