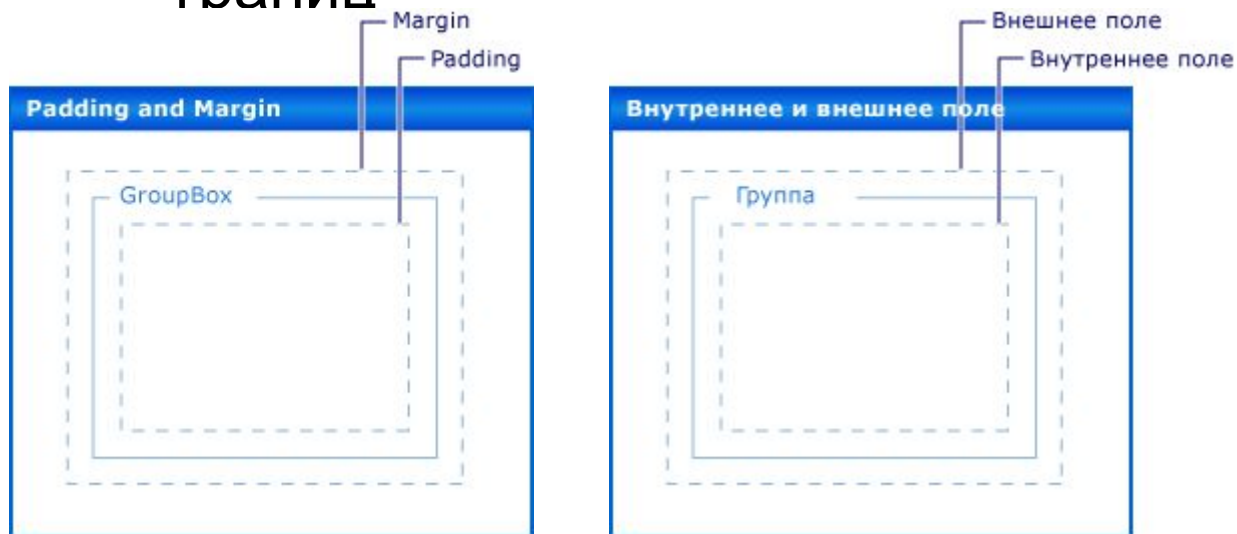


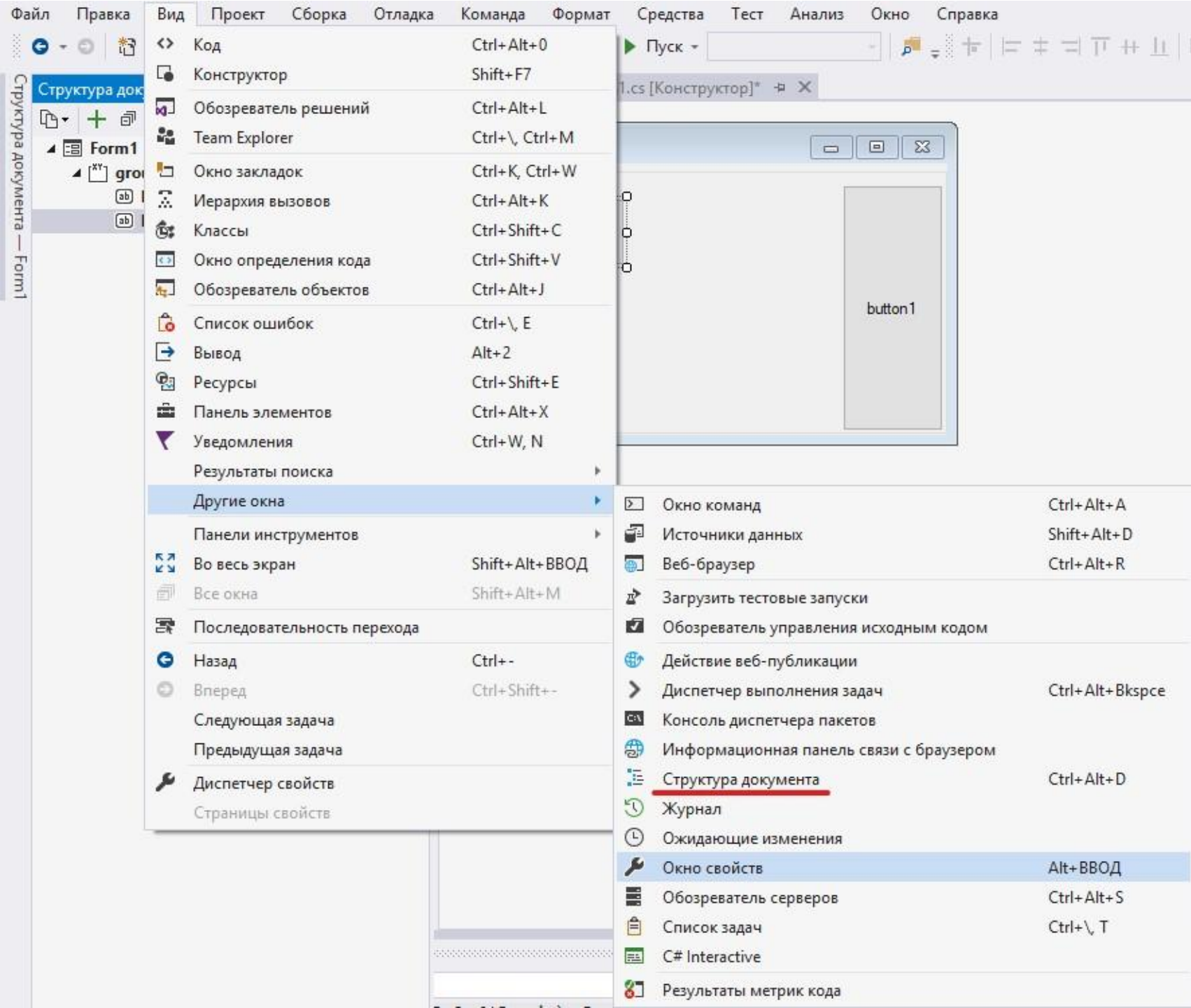
# Свойства для размещения и размера элементов

- **Size**: очевидно размеры
- **Margin**: отступ вокруг элемента управления, расстояние до других ЭЛ-ОВ
- **Padding**: отступ внутри элемента управления, расстояние до его границ



Сам по себе Margin не изменяют отступы(что странно), но при перетаскивании изменяется длина линий привязки

А Padding видно как работает на полях с текстом(label)

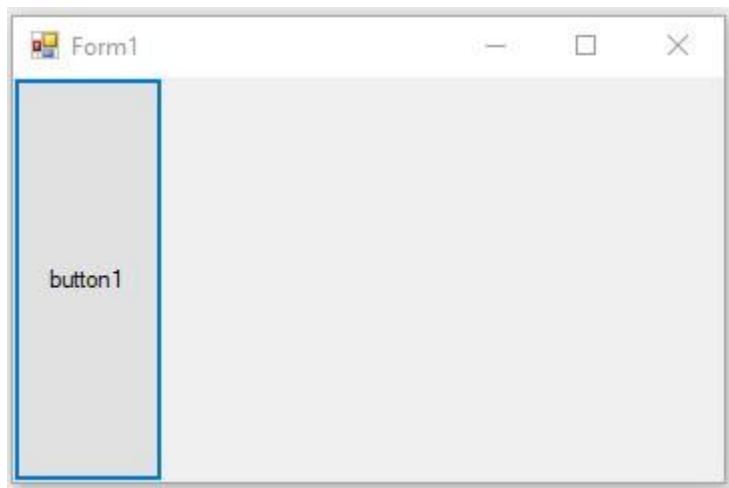


**полезная  
вкладка  
чтобы  
посмотреть  
все  
элементы на  
форме:**

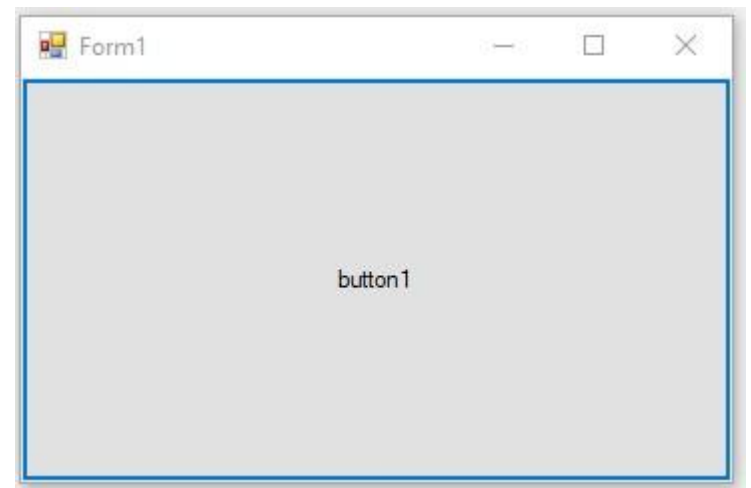
**Anchor:** с каких сторон элемент будет прицепляться к внешнему контейнеру при изменении размеров экрана (если задать со всех сторон, то он будет растягиваться во весь объем). Можно задать несколько сторон

**Dock:** почти то же самое, но более интуитивно, какую сторону будет занимать элемент(только одну)

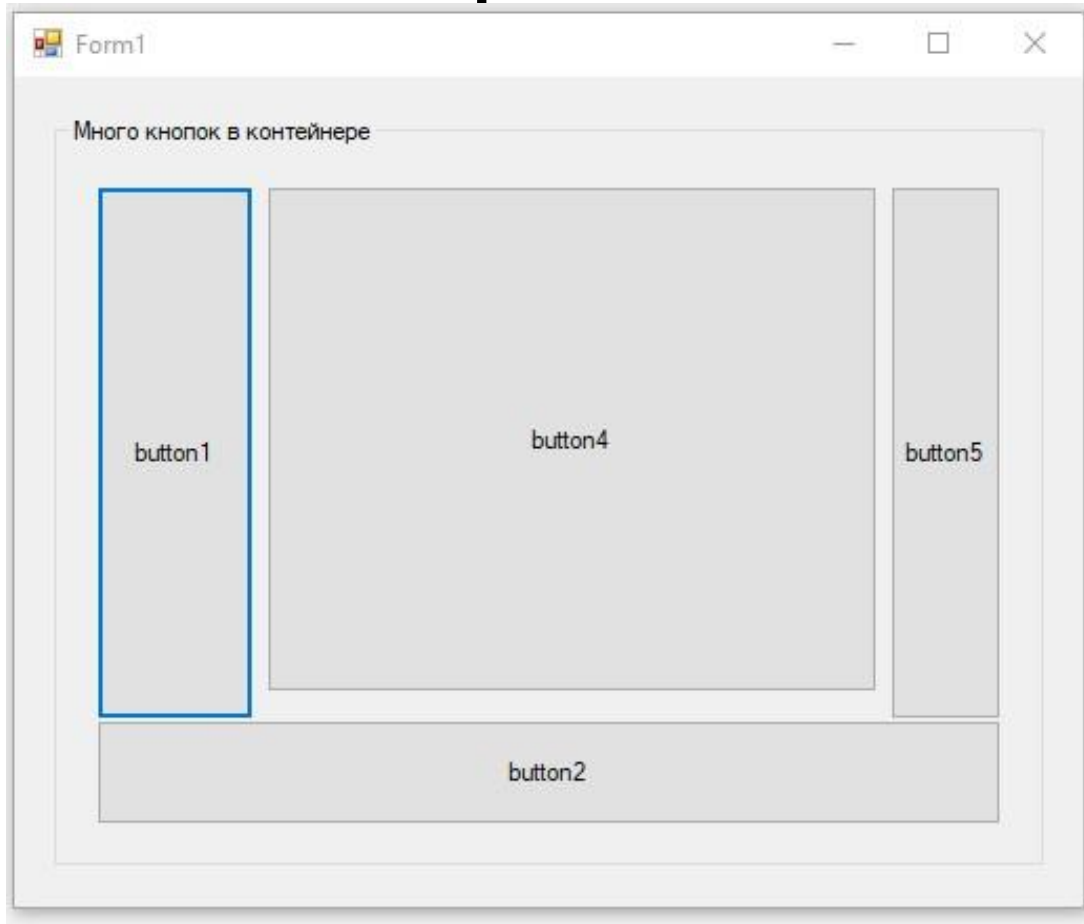
Dock=Left



Dock=Fill

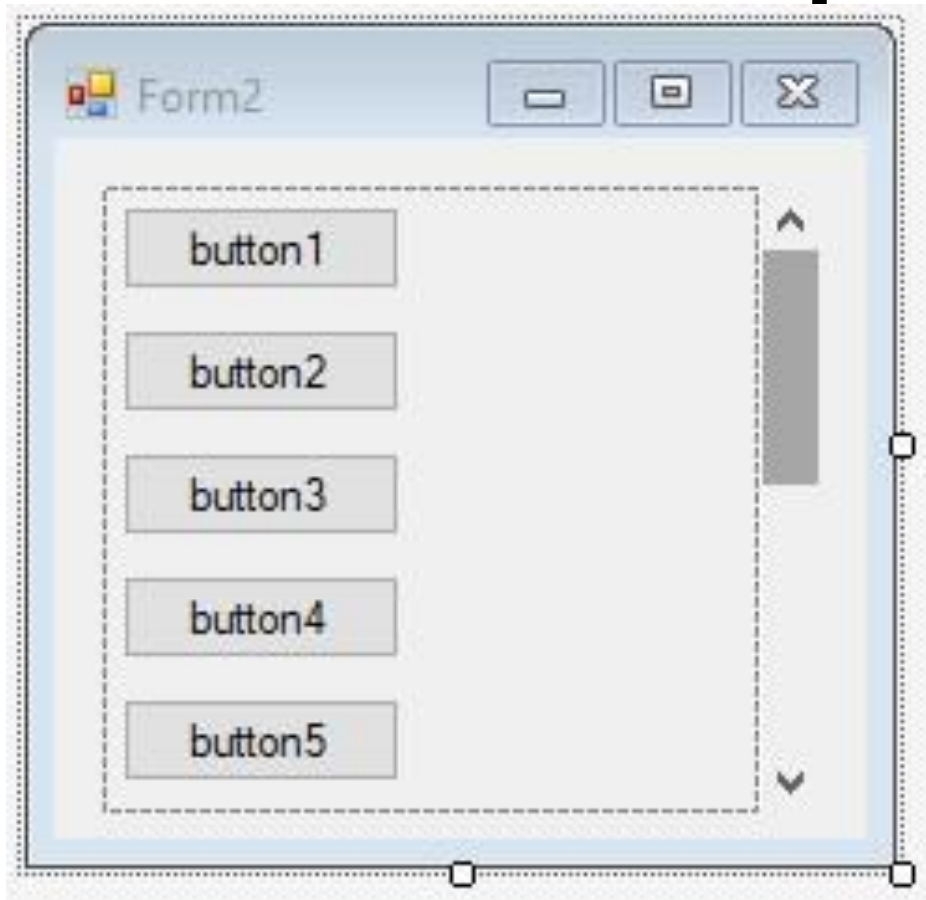


Но разделять экран на несколько частей только с помощью этих свойств не удобно, поэтому существуют различные контейнеры:

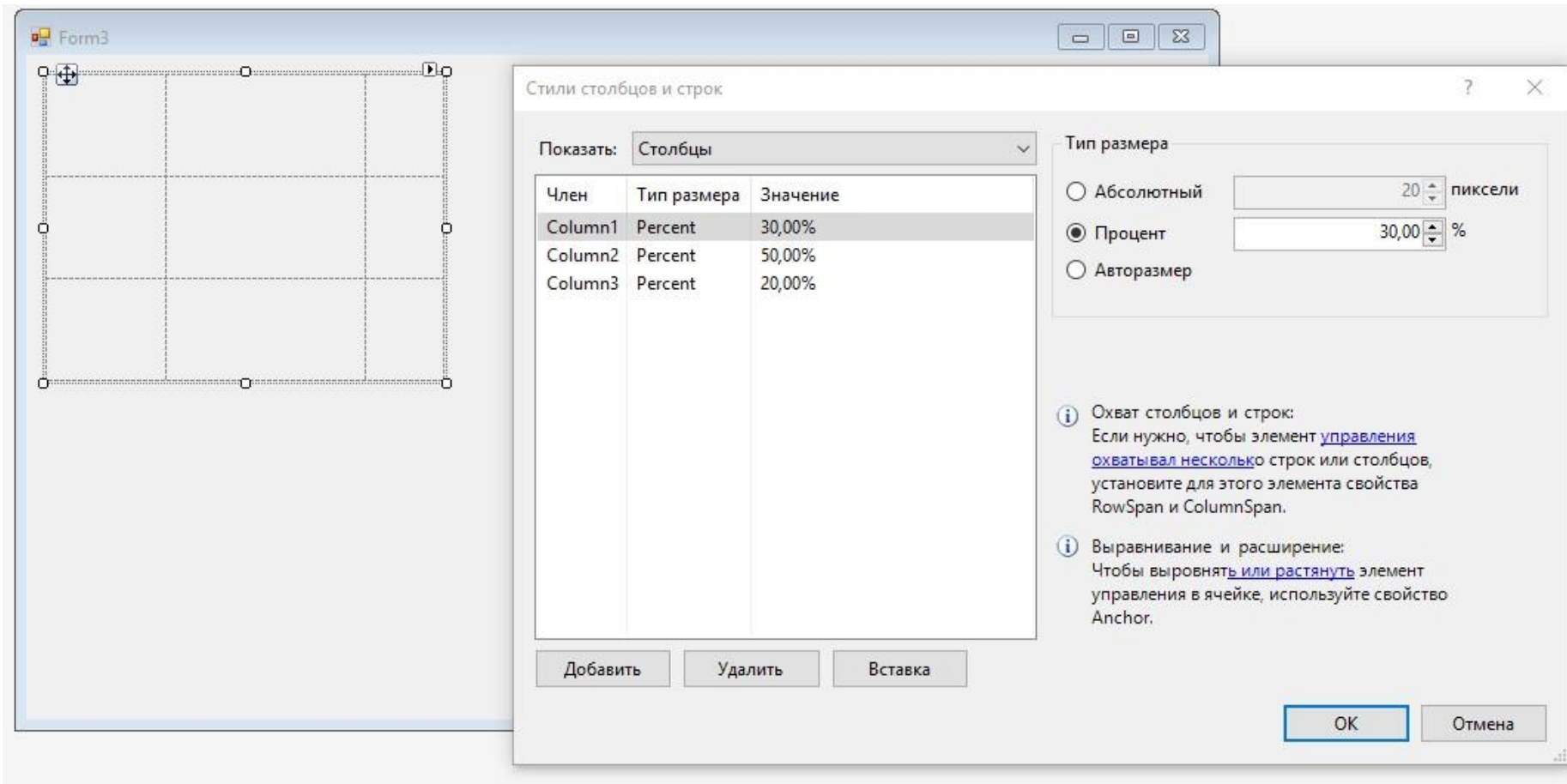


**GroupBox:** контейнер с границей и заголовком

**Panel:** такой же контейнер для объединения элементов, только без заголовка и в нем можно установить возможность прокрутки

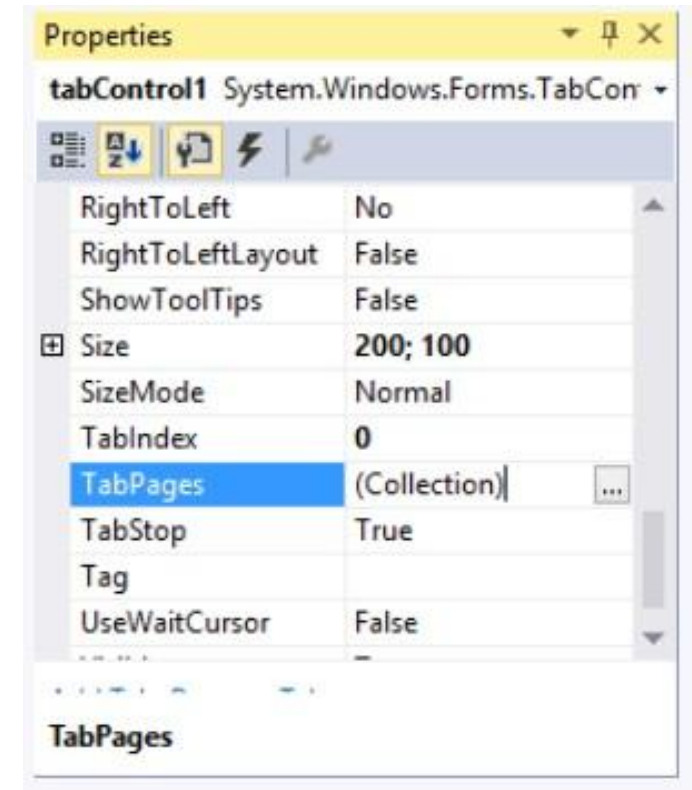
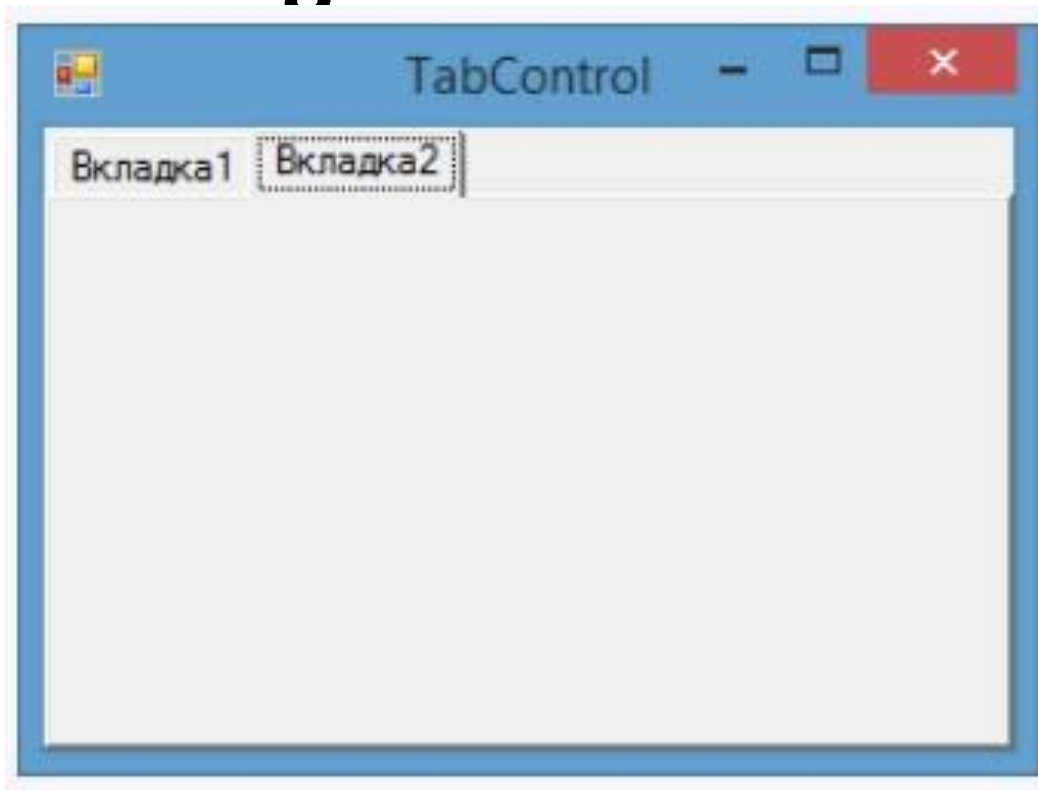


**TableLayoutPanel:** располагает элементы в виде таблицы. Если нам хочется поместить в ячейку более одного элемента, то в эту ячейку добавляется другой компонент `TableLayoutPanel`. Его можно настроить с помощью ПКМ->Правка строк и столбцов.



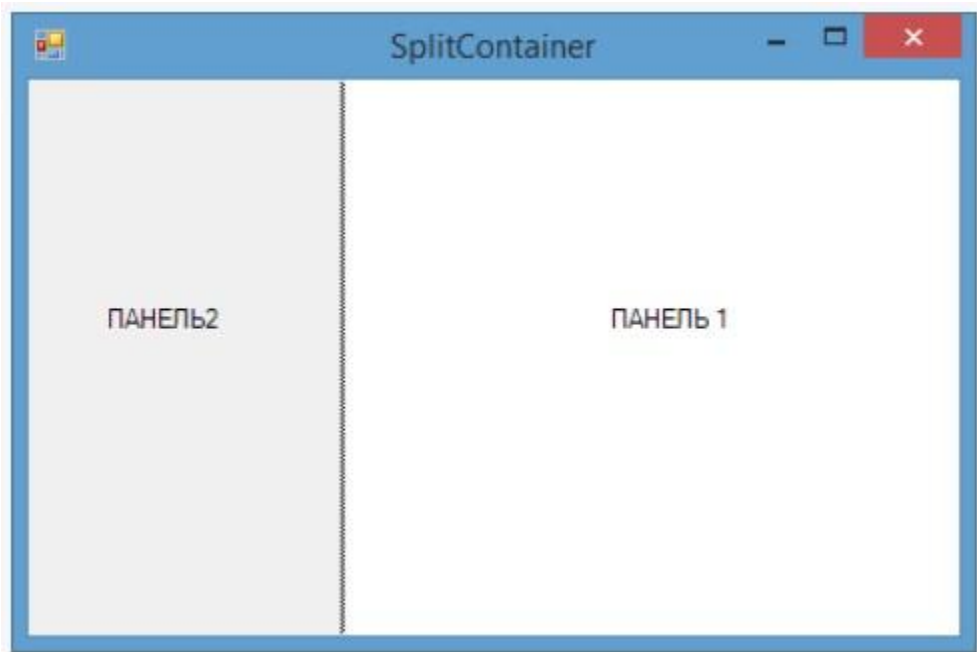
И уже в эти ячейки можно удобно размещать элементы с Dock=Fill, чтобы все выглядело красиво и при изменении размера не поехало в стороны

**TabControl:** несколько вкладок на одной форме, каждая вкладка это простой контейнер. Чтобы настроить (добавить, удалить) вкладки используем свойство **TabPage**





**SplitContainer:** просто две разделенные панели. Изменяя положение разделителя, можно изменить размеры этих панелей.



Также можно сделать вертикальный разделитель, поменяв свойство Orientation

И главное понять, что все операции вы можете выполнять также в коде, например добавлять в него динамически новые элементы через метод `.Add()`, изменять его размеры, количество строк и все другие параметры почти так же просто, через методы этих классов. Например:

## Изменение размеров строк и столбцов в TableLayoutPanel

```
1 tableLayoutPanel1.RowStyle[0].SizeType = SizeType.Percent;  
2 tableLayoutPanel1.RowStyle[0].Height = 40;  
3  
4 tableLayoutPanel1.ColumnStyle[0].SizeType = SizeType.Absolute;  
5 tableLayoutPanel1.ColumnStyle[0].Width = 50;
```

## Добавление кнопки в ячейку таблицы 2,2

```
1 Button saveButton = new Button();  
2 // добавляем кнопку в следующую свободную ячейку  
3 tableLayoutPanel1.Controls.Add(saveButton);  
4 // добавляем кнопку в ячейку (2,2)  
5 tableLayoutPanel1.Controls.Add(saveButton, 2, 2);
```

```
1 //добавление вкладки  
2 TabPage newTabPage = new TabPage();  
3 newTabPage.Text = "Континенты";  
4 tabControl1.TabPages.Add(newTabPage);
```