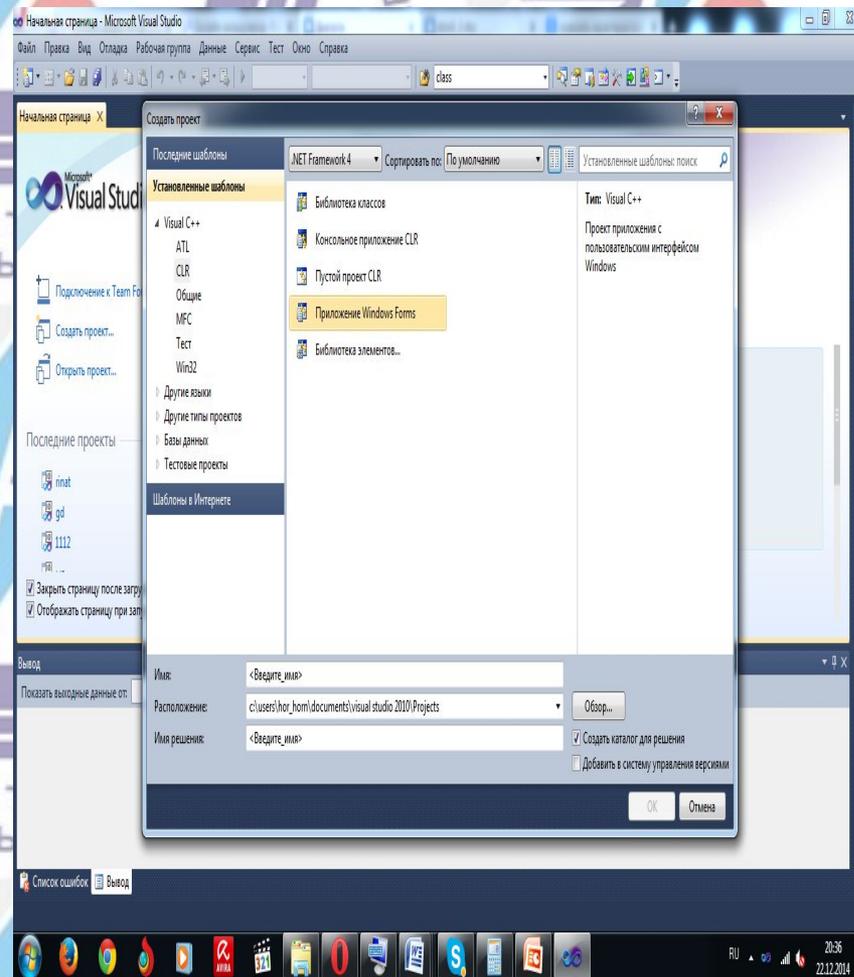


Презентация по информатике  
«работа с windows form  
application»

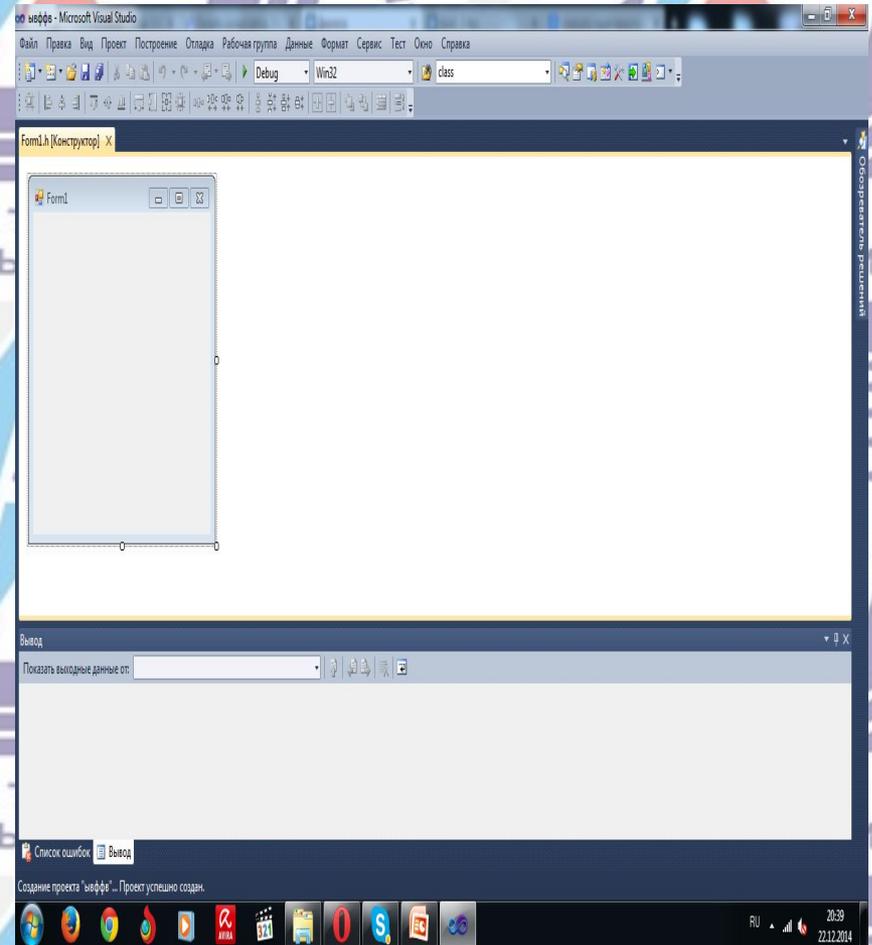
# Создание проекта

Для того, что бы  
создать форму в  
меню создание  
проекта выберите  
создать  
«приложение  
Windows Form».



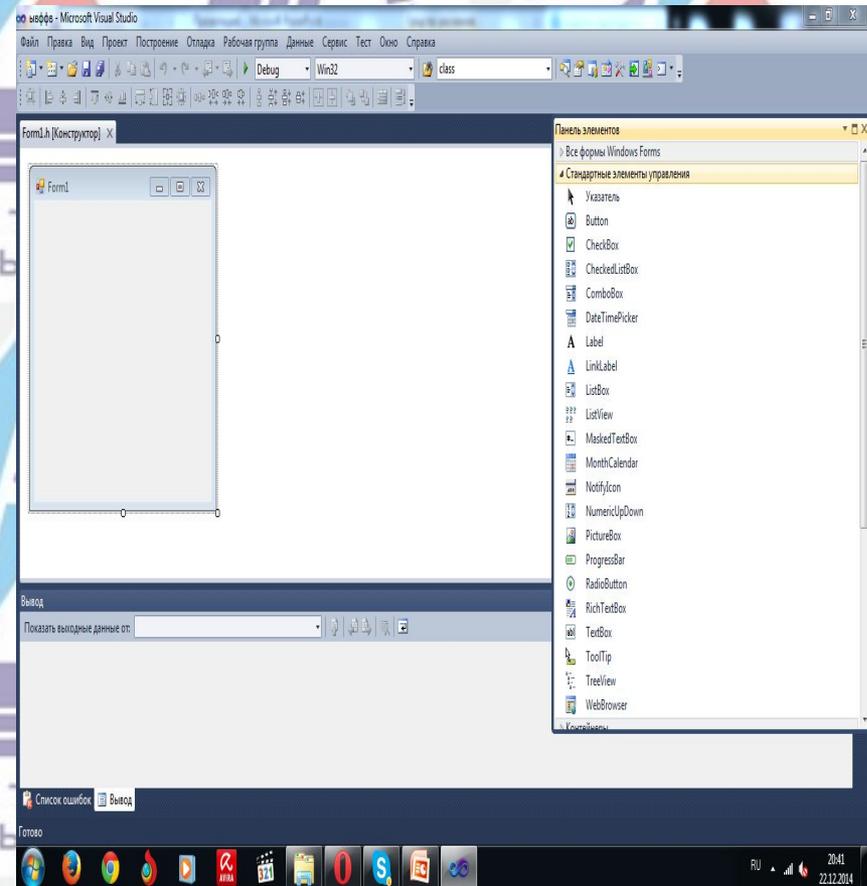
# Рабочая область

Перед вами появится окно конструктора вашей формы. На которую вы будете добавлять элементы управления программы.



# Панель Элементов

Нажатием **Ctrl+alt+x** вы вызовете «Панель элементов». На которой располагаются элементы которые будут использоваться в вашей будущей программе.



# Button

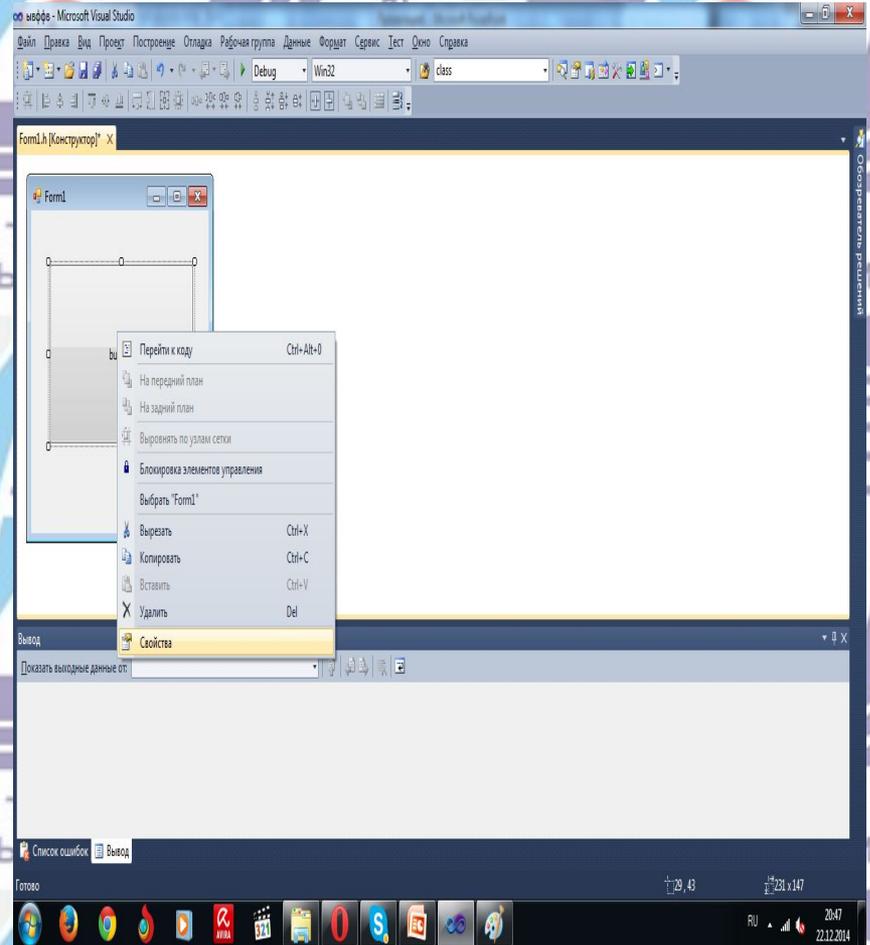
Разберем самые основные элементы

Button(кнопка)-элемент управления при нажатии на неё происходит программно связанное с этим нажатием действие либо событие. Множество возможных разновидностей сложных кнопок бесконечно. Сложные кнопки могут быть фиксируемыми (сохранять текущее состояние после завершения нажатия), реагировать на быстрые двойное, тройное и более нажатия (клики), включать многофазные счётные триггеры и другие более сложные устройства с большим количеством состояний.



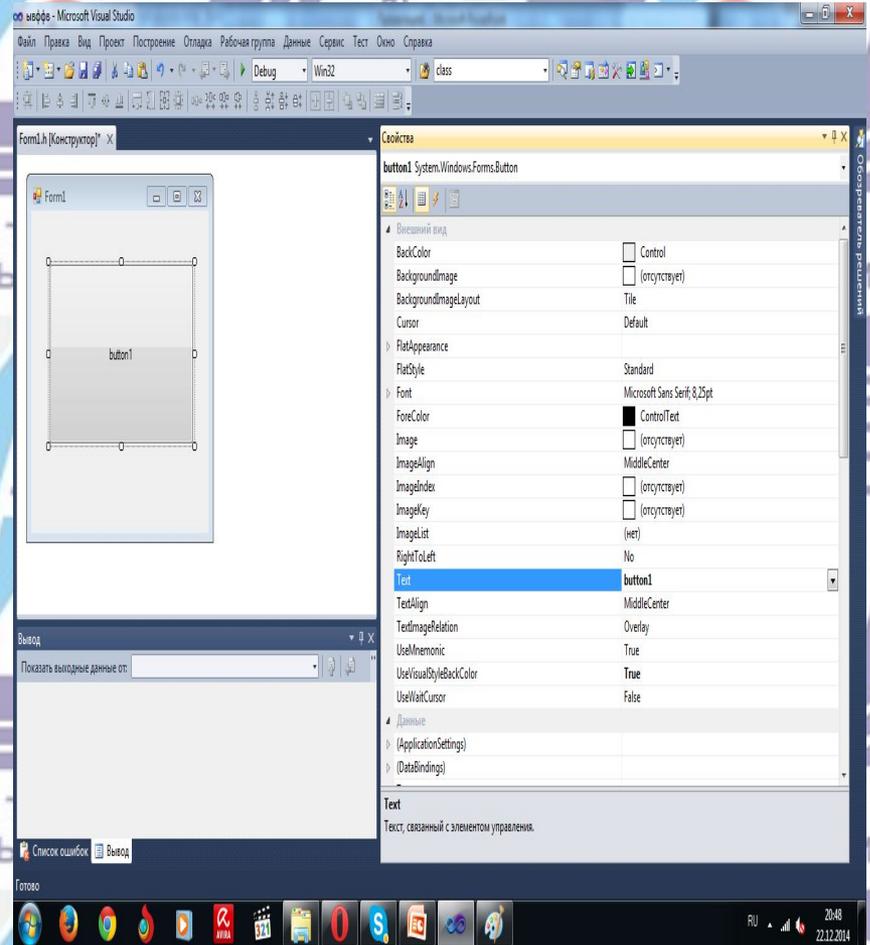
# Button

Нажимая на кнопку правой клавишей мы можем вызвать меню свойств. Или скопировать, вырезать и удалить кнопку.

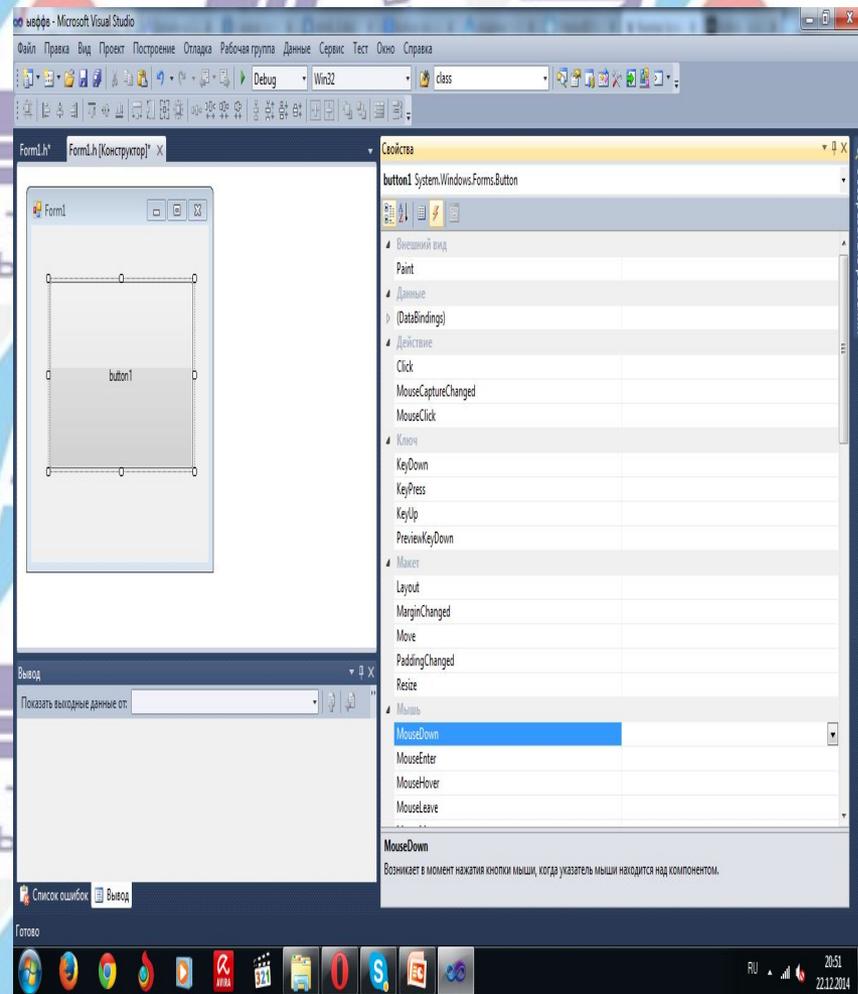


# Button

В свойствах вы можете изменить название кнопки, размер, цвет, изображение, шрифт надписи на кнопке, расположение текста и . т. п.

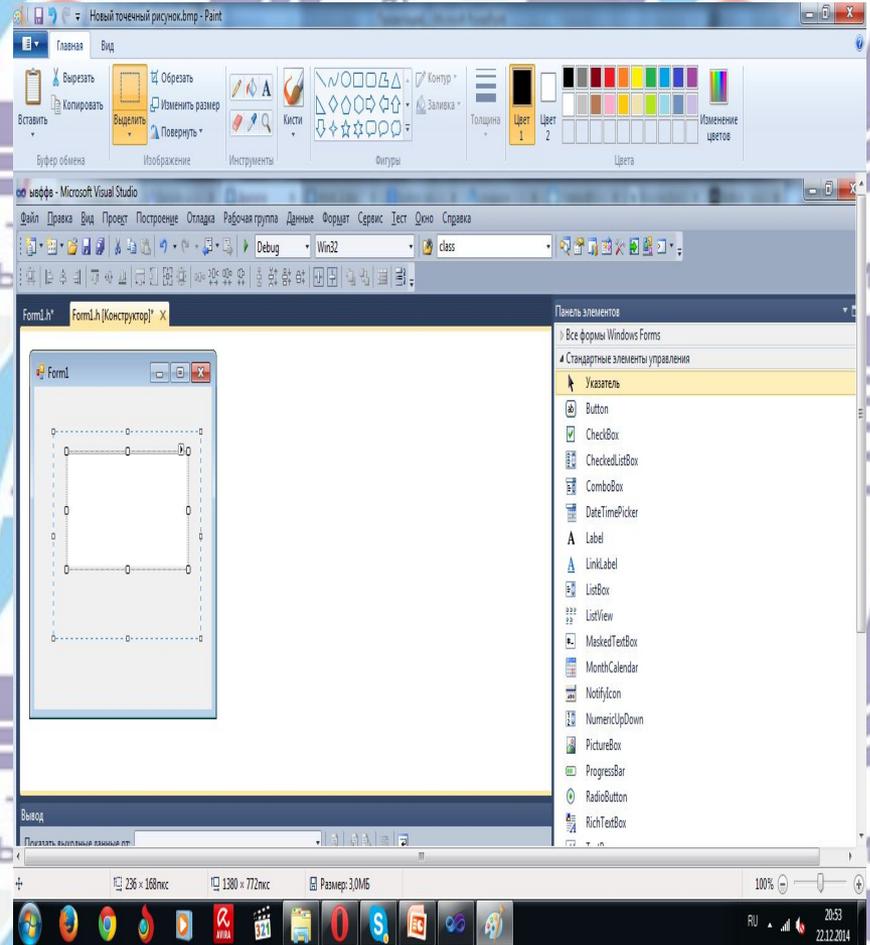


Если вы нажмете на значок молнии в свойствах кнопки. То перейдете на вкладку «Действий». Где выбирается по какому действию с кнопкой будет выполняться то или иное действие в программе.



# TextBox

Следующий наш элемент TextBox(поле редактирования). Это поля куда вы можете вводить данные которые будут использоваться у вас в коде или выводить.



# TextBox

В нем как и в Button и других элементах есть панель свойств и действий. И отвечают они все так же за графическое отображение элемента и на какое действие он будет реагировать.

**Важный совет для того, что бы сделать TextBox многострочным в свойстве Multiline поставте значени True.**



Multiline	True
PasswordChar	
ReadOnly	False
RightToLeft	No

# ComboBox

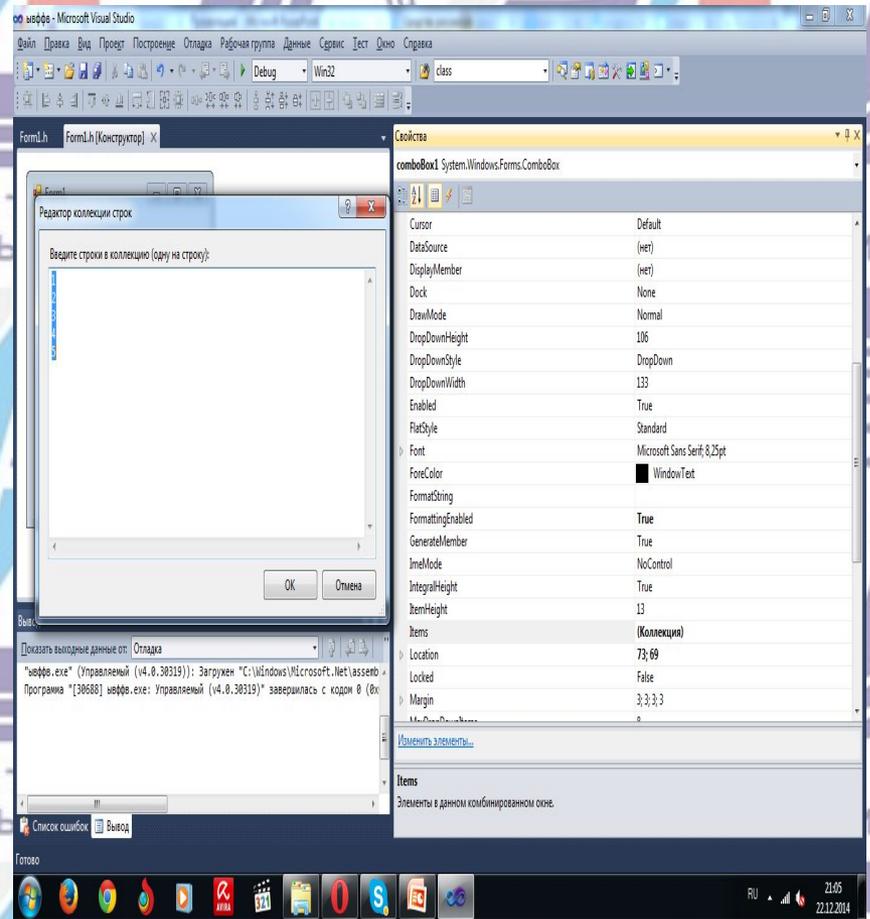
Третий наш элемент  
ComboBox ( список ,  
выпадающие список ).  
Сочетание  
выпадающего списка  
(раскрывающегося при  
щелчке мыши) и  
однострочного  
текстового поля,  
которое позволяет  
пользователю ввести  
значение вручную или  
выбрать из списка.



# ComboBox

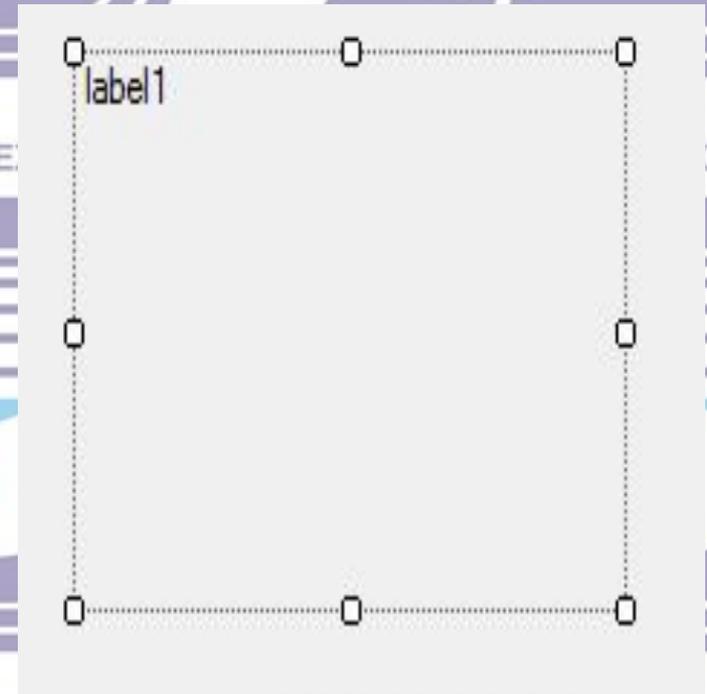
Что важно знать про него? У него так же есть свойство Multiline.

И значения которые можно выбирать вы заносите в Свойстве –Items.



# label

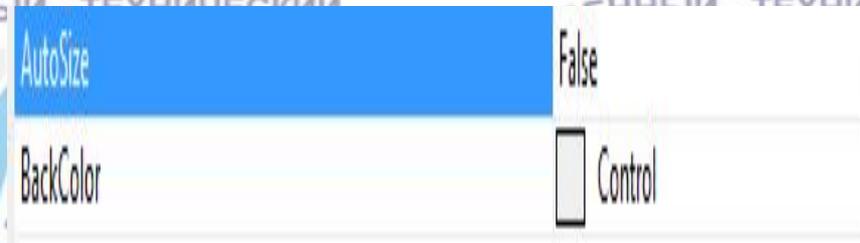
И наконец наш последний на сегодня элемент, но очень важный label(надпись).  
Используется для того что бы создать надпись на форме.



label

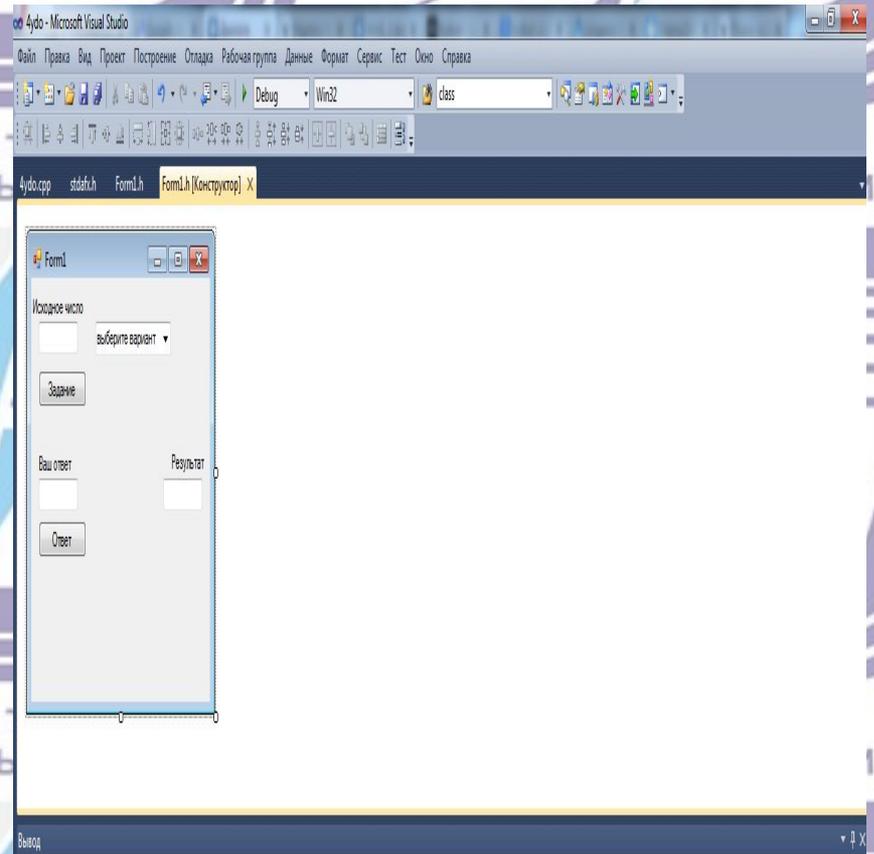
Текст надписи  
задается в  
Свойствах-Text.

Важно знать если  
хотите увеличить  
размер Label в  
Свойствах-Autosize  
поставьте значение  
false.



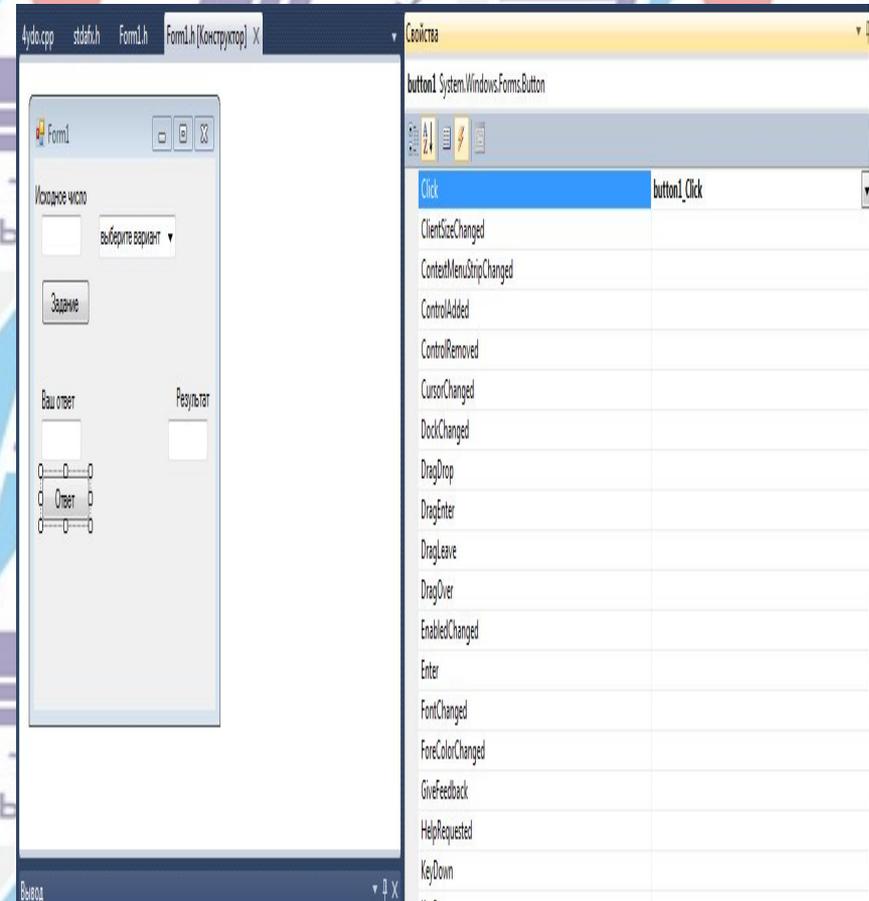
# Работа с кодом

Рассмотрим работу с элементами с кодом программы. Как видите на форме добавлены элементы управления.



# Работа с кодом

Щелкаем на Button 1  
заходим в Действия,  
и выбираем нужное.  
В нашем случае это  
одинарный клик.



# Работа с кодом

Вас сразу перенесет на страницу где вы будете писать свой код.

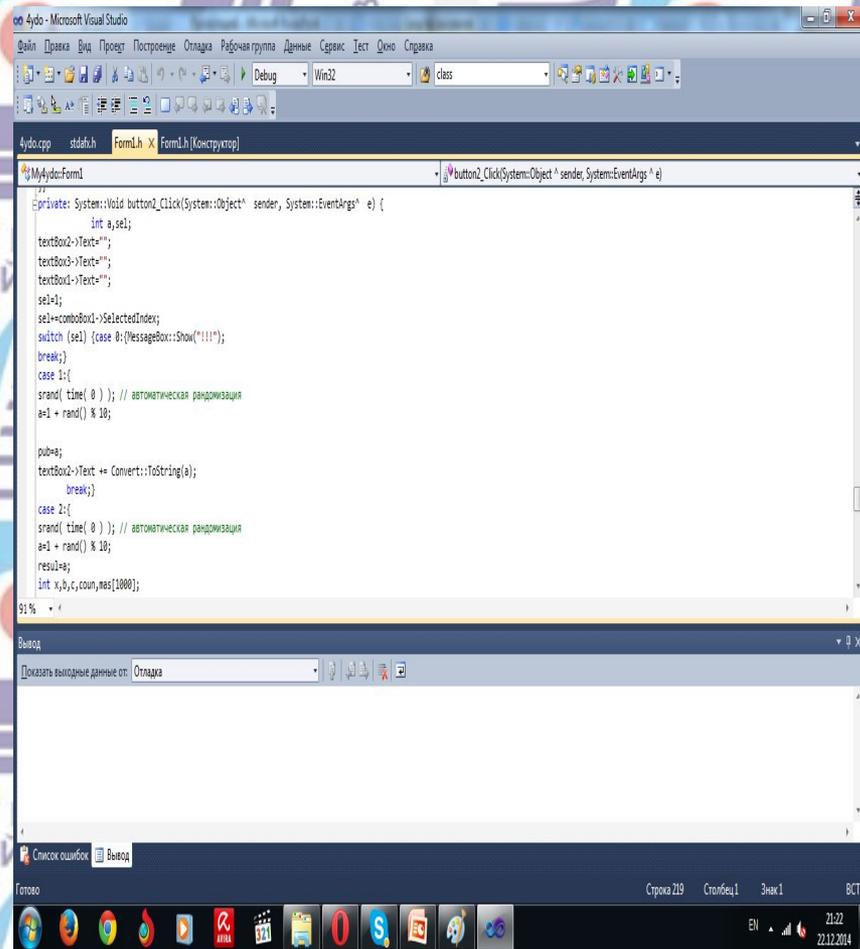
Void button\_click

Это наша функция где мы пишем код, все как и с консольными приложениями.

```
private: System::Void button1_Click(System::Object^ sender, System::EventArgs^ e) {  
    }  
};
```

# Работа с кодом

Можно увидеть, что  
прямо в клавише я  
объявлял  
переменные.



The screenshot shows the Microsoft Visual Studio IDE. The main window displays a C++ code file named 'MyMyForm1'. The code is a private method 'button\_Click' that handles a button click event. The code includes variable declarations and assignments within the function body, such as 'int a,sel;', 'textBox2->Text="";', and 'sel=1;'. The code also features a switch statement and a loop. The status bar at the bottom indicates 'Страница 219', 'Столбец 1', 'Знак 1', and 'ВСТ'. The system tray shows the date and time as '22.12.2014' and '21:22'.

```
private: System::Void button_Click(System::Object^ sender, System::EventArgs^ e) {  
    int a,sel;  
    textBox2->Text="";  
    textBox3->Text="";  
    textBox1->Text="";  
    sel=1;  
    sel+=comboBox1->SelectedIndex;  
    switch (sel) {case 0: MessageBox::Show("!!!");  
    break;}  
    case 1: {  
    srand( time( 0 ) ); // автоматическая рандомизация  
    a=1 + rand() % 10;  
  
    pub=a;  
    textBox2->Text += Convert::ToString(a);  
    break;}  
    case 2: {  
    srand( time( 0 ) ); // автоматическая рандомизация  
    a=1 + rand() % 10;  
    result+=a;  
    int x,y,c,con,mas[1000];
```

# Работа с кодом

А вот тут мы присваиваем переменной значение из TextBox. Значение в TextBox изначально типа String так, что не забудьте перевести его в нужный вам формат.

```
a=pub;  
otvet = Convert.ToInt32(textBox3->Text);  
coun=0;
```

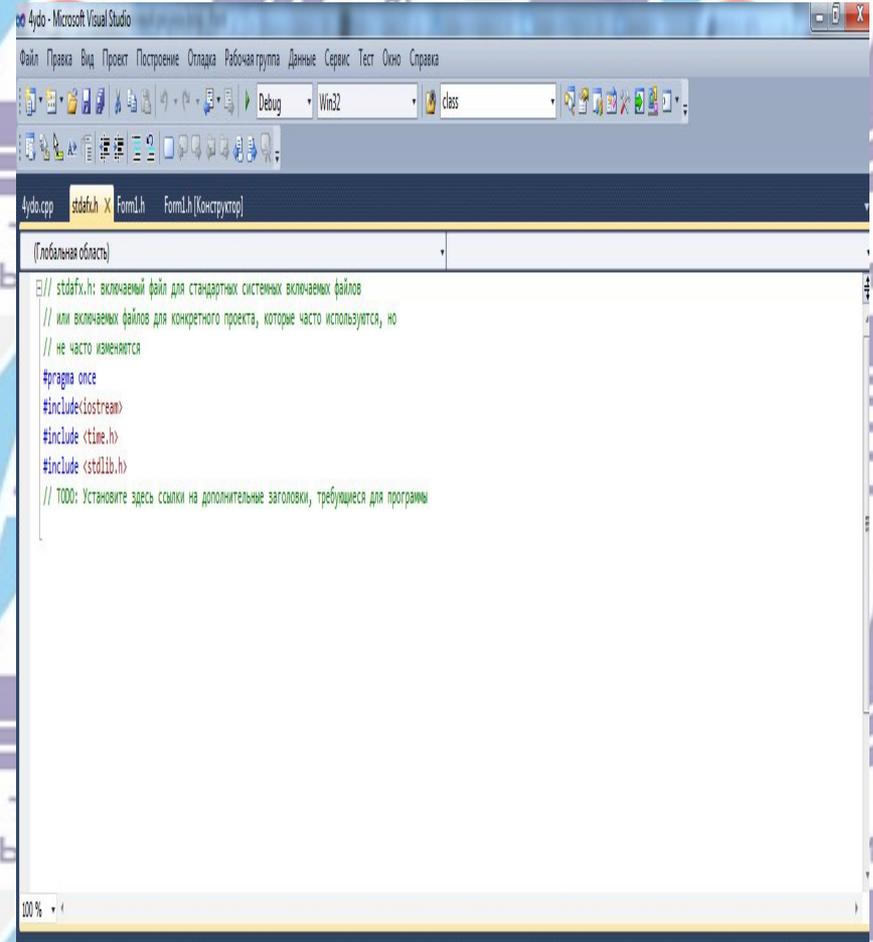
# Работа с кодом

Здесь мы присваиваем переменной значение из ComboBox. С помощью метода SelectedIndex. Метод работает так: каждой строчке комбабокса соответствует значение. Первое значение = -1.

```
sel+=comboBox1->SelectedIndex;  
switch (sel) {case 0:{MessageBox::Show("выберите тип перевода");
```

# Работа с кодом

И самое главное.  
Прописывать  
используемые вами  
библиотеки следует  
в файле `stdafx.h`  
после `#pragma once`.



```
4\jdo.cpp  stdafx.h  Form1.h  Form1.h (Конструктор)
(Глобальная область)
// stdafx.h: включенный файл для стандартных системных включаемых файлов
// или включаемых файлов для конкретного проекта, которые часто используются, но
// не часто изменяются
#pragma once
#include <iostream>
#include <time.h>
#include <stdlib.h>
// TODO: Установите здесь ссылки на дополнительные заголовки, требующиеся для программы
```

# Рабочая программа

