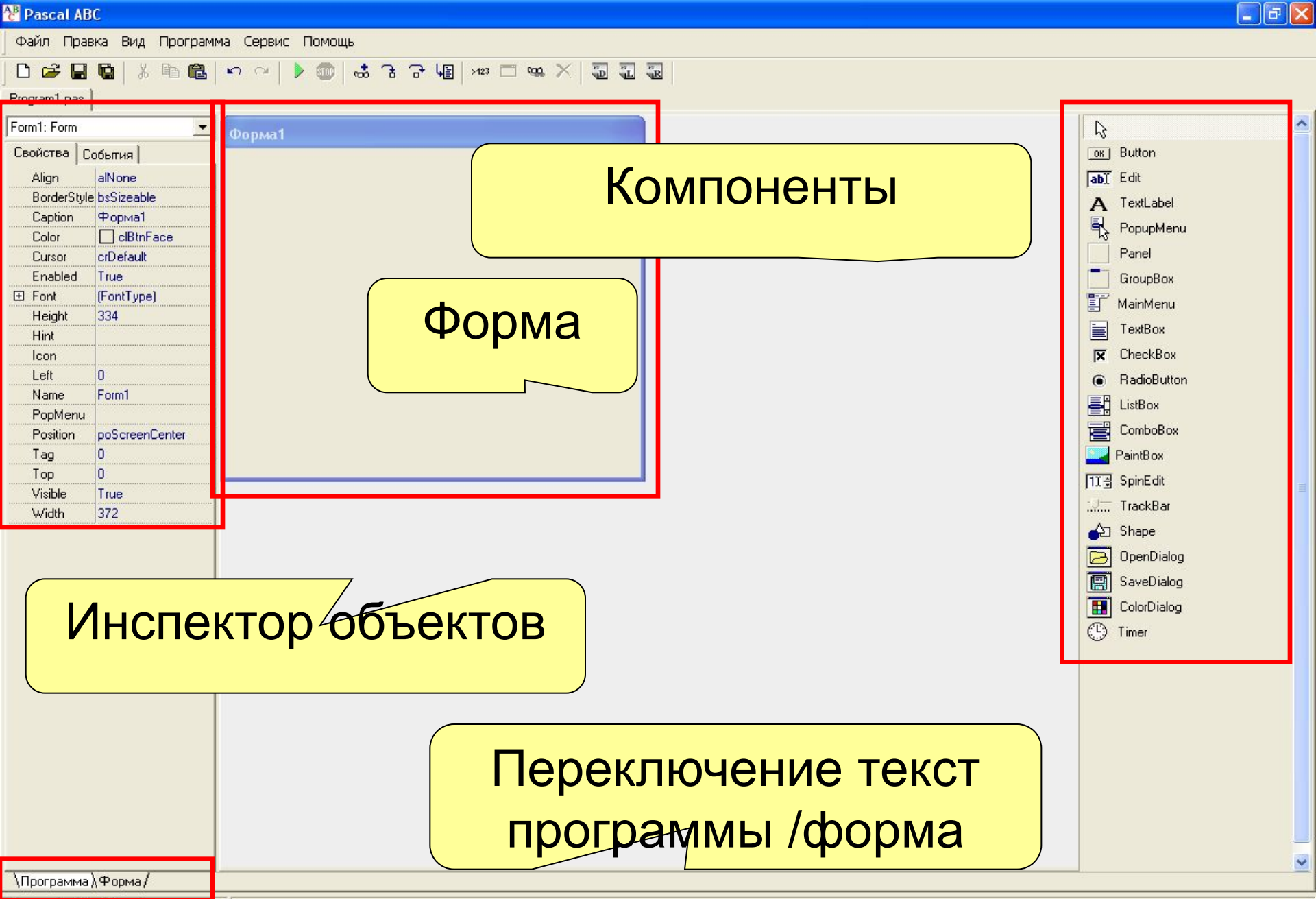


# Объектно-ориентированное программирование

Практические работы в Pascal ABC



Компоненты

Форма

Инспектор объектов

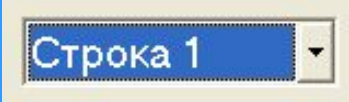
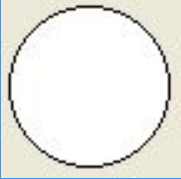


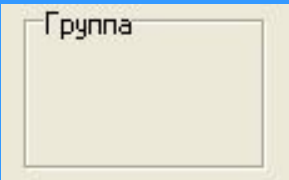
Переклечение текст программы /форма

\Программа\Форма\






## Визуальные компоненты

	<p><b>Button (Кнопка)</b> Обычная кнопка предназначена для выполнения какого-либо кода по наступлению события нажатия на кнопку пользователем</p>
	<p><b>Edit (Поле для ввода)</b> Служит для отображения текстовой информации, которую пользователь может изменять. Удобно применять для ввода или редактирования какой-либо текстовой информации</p>
	<p><b>Label (Надпись)</b> Служит для простого отображения текстовой информации, которую пользователь не может редактировать во время работы приложения</p>
	<p><b>CheckBox (Флажок)</b> Представляет опцию, благодаря которой пользователь может переключать положение (вкл./выкл.)</p>
	<p><b>TextBox (Многострочное поле)</b> Работает как и поле для ввода, только позволяет вводить (редактировать) несколько строк текста</p>

## Визуальные компоненты

	<p><b>ComboBox (Выпадающий список)</b> Является как бы комбинацией списка и поля для ввода. Пользователь может ввести самостоятельно текст в поле, либо выбрать нужный из предоставленного списка</p>
	<p><b>Shape (Геометрическая фигура)</b> Позволяет отображать произвольные геометрические фигуры. Например, эллипс, треугольник, прямоугольник и другие</p>
	<p><b>RadioButton (Переключатель)</b> Данный переключатель работает обычно в группе с другими. Если на форме располагается несколько таких переключателей, то только один из них может принимать включенное положение</p>
	<p><b>ListBox (Список)</b> Предназначен для вывода нескольких строк текста. Пользователь может выбрать любую из этих строк простым нажатием на ней</p>
	<p><b>GroupBox (Группа)</b> Создает контейнер, который может содержать внутри себя компоненты. Эти компоненты объединяются в одну группу.</p>

## Не визуальные компоненты

 OpenFileDialog1	<p><b>OpenDialog</b> компонент* предназначен для поддержки операции открытия файлов. При обращении к этому компоненту вызывается стандартное диалоговое окно открытия файла.</p> <p><i>*(место размещения которого на форме не имеет значения для пользователя так как он увидит не сам компонент, а меню или диалог , с генерированные им),</i></p>
 SaveDialog1	<p><b>SaveDialog</b> предназначен для поддержки операции сохранения файлов. При обращении к этому компоненту вызывается стандартное диалоговое окно сохранения файла.</p>
 ColorDialog1	<p><b>ColorDialog</b> предназначен для поддержки операции выбора цвета. При обращении к этому компоненту вызывается стандартное диалоговое окно выбора цвета.</p>
 Timer1	<p><b>Timer (Таймер)</b> позволяет вводить необходимые задержки между выполнением тех или иных действий</p>
 MainMenu1	<p><b>MainMenu (Главное меню)</b> предназначенный для вывода главного меню на форме</p>

Свойство	Назначение
Color	цвет компонента
Caption	надпись на компоненте
Font	сложное свойство, определяющее тип шрифта надписи
Visible	логическое свойство, определяющее видимость компонента
Enabled	логическое свойство, определяющее доступность компонента
Top	местоположение левого верхнего угла компонента по вертикали
Left	местоположение левого верхнего угла компонента по горизонтали
Height	высота компонента
Width	ширина компонента

# Изменение свойств объекта

1 способ:

Программно

**Button1.Caption:='Выход';**

Как изменить  
надпись на кнопке?



Выход

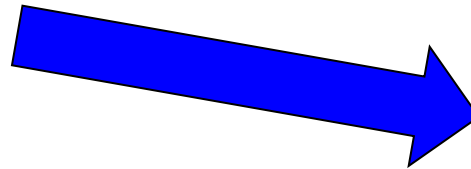
2 способ:

Инспектор объектов

Свойства	События
Align	alNone
Caption	Выход
Cursor	crDefault
Enabled	True
⊕ Font	(FontType)
Height	33
Hint	
Left	104
Name	Button5
PopupMenu	
Tag	0
Top	96
Visible	True
Width	97

Поле ввода Edit1  
(String)

123



Оперативная память

**A:=StrToInt(Edit1.Text);**

123

Переменная А  
(Integer)

**B:=StrToFloat(Edit1.Text);**

123

Переменная В  
(Real)

**C:=Edit1.Text;**

'123'

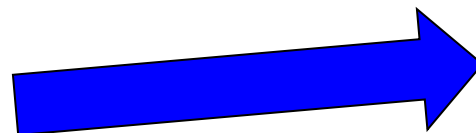
Переменная С  
(String)

Внимание! Функция **StrToInt** – не округляет число введенное в поле ввода.



Поле ввода Edit1  
(String)

123



Оперативная память

Переменная A  
(Integer)

123

`Edit1.Text:=IntToStr (A);`

Переменная B  
(Real)

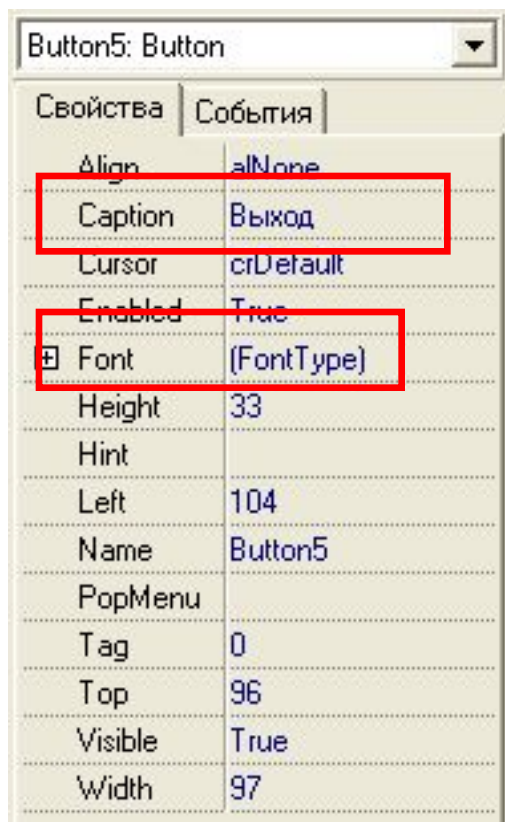
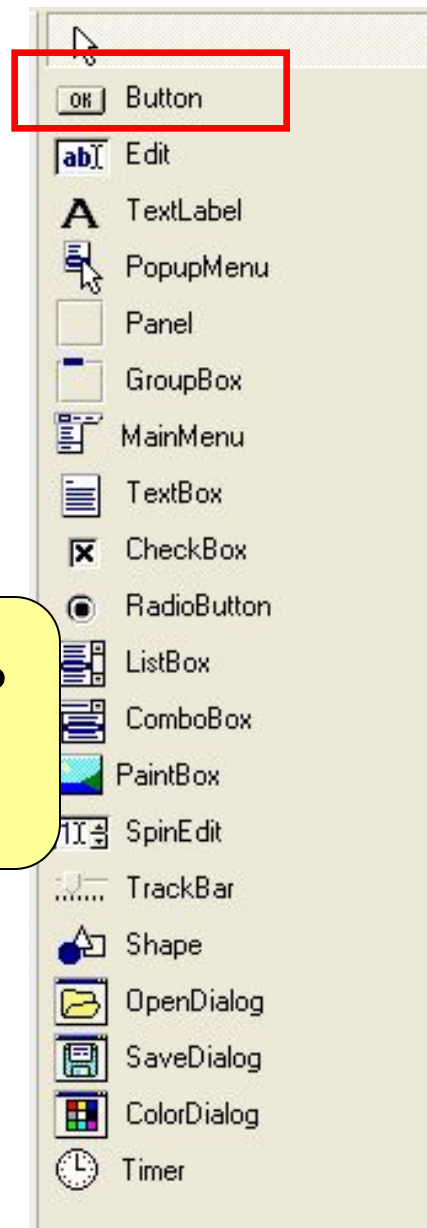
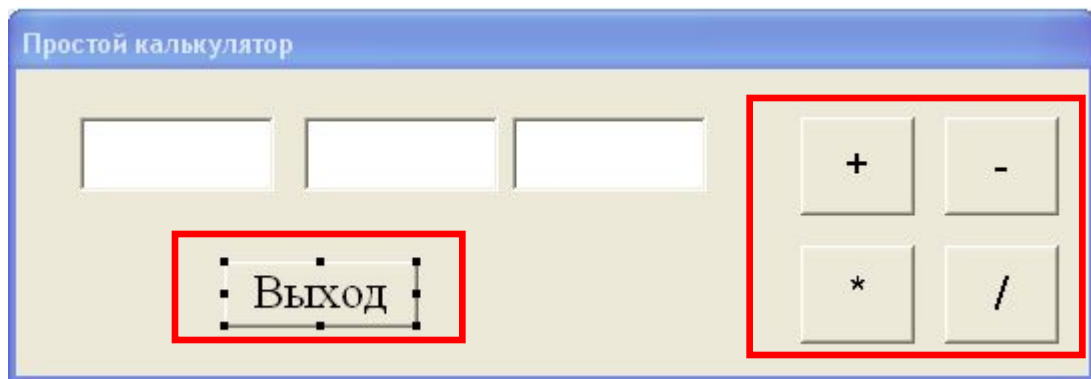
123

`Edit1.Text:=FloatToStr (B);`

Переменная C  
(String)

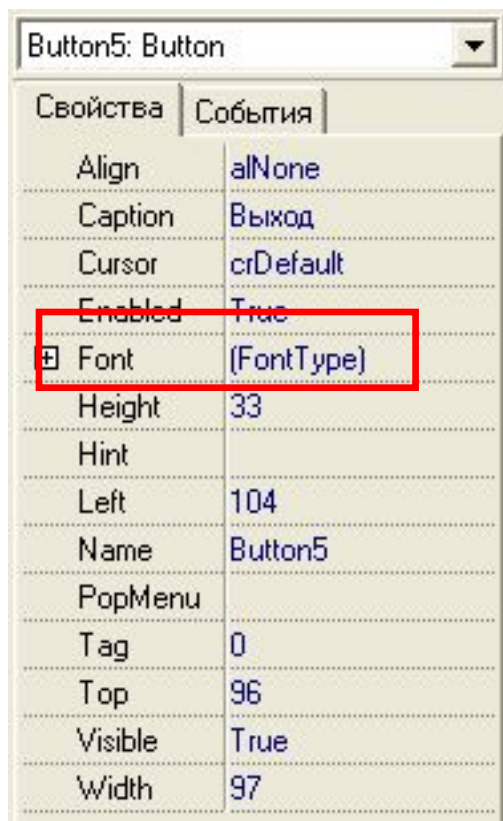
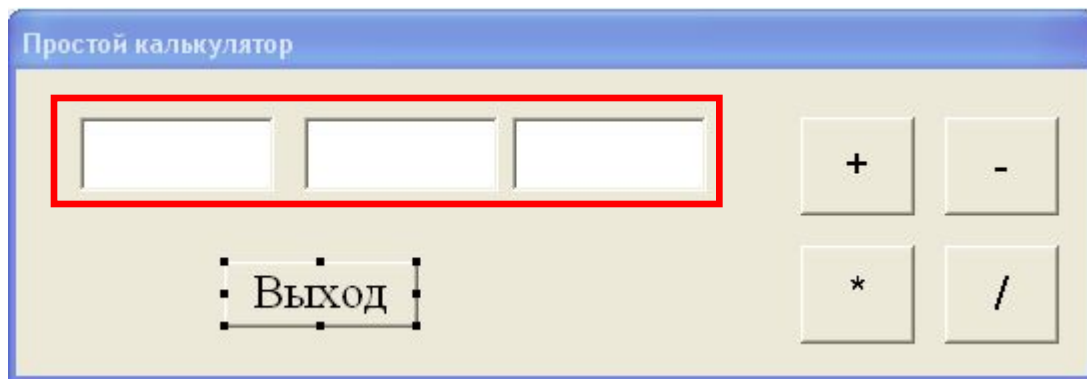
'123'

`Edit1.Text := C;`



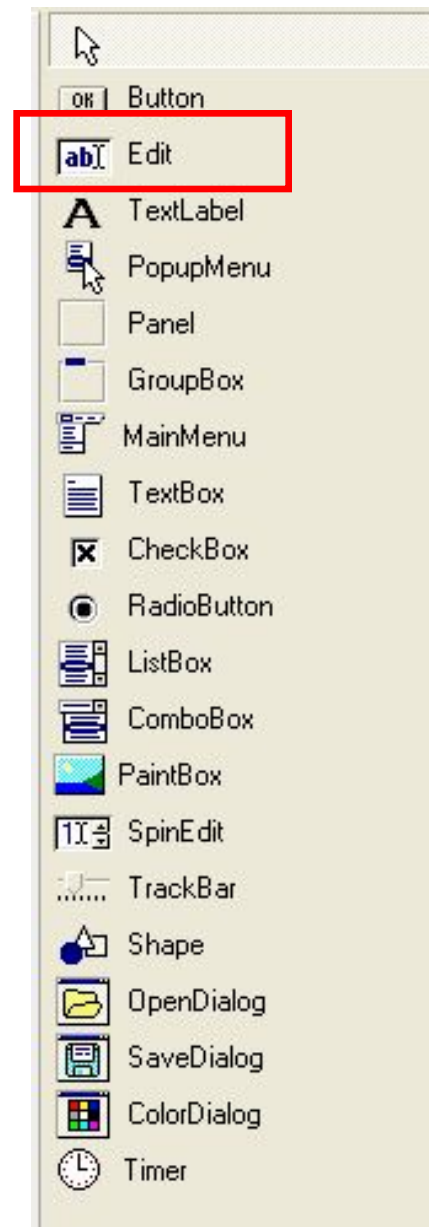
Разместим на  
Изменим надпись  
на кнопках

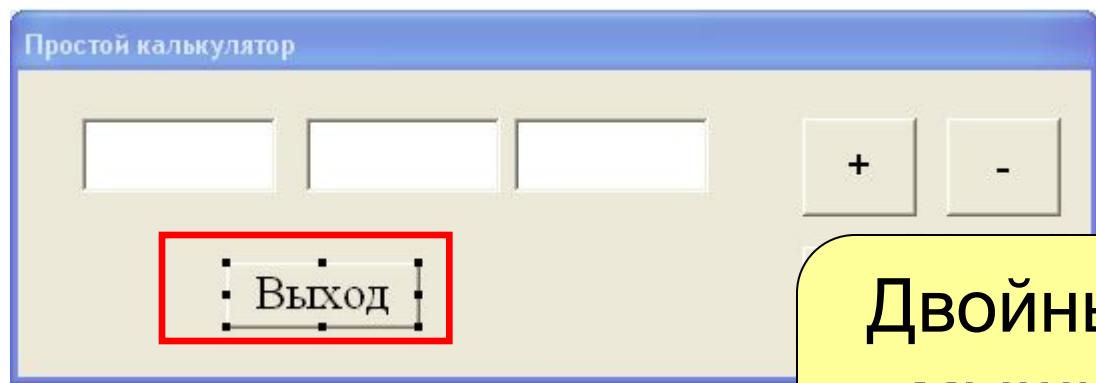
Изменим размер  
надписи



Разместим на  
форме  
компоненты  
поле для ввода  
(Edit)

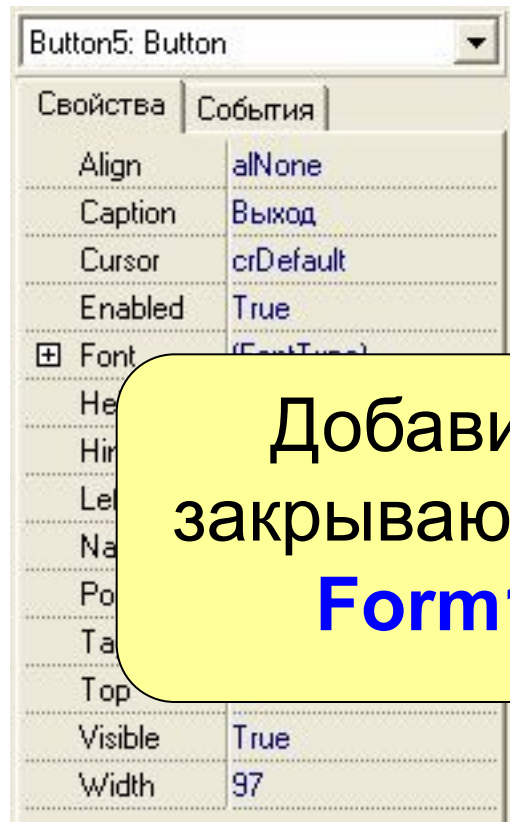
Изменим размер  
символов в поле  
ввода



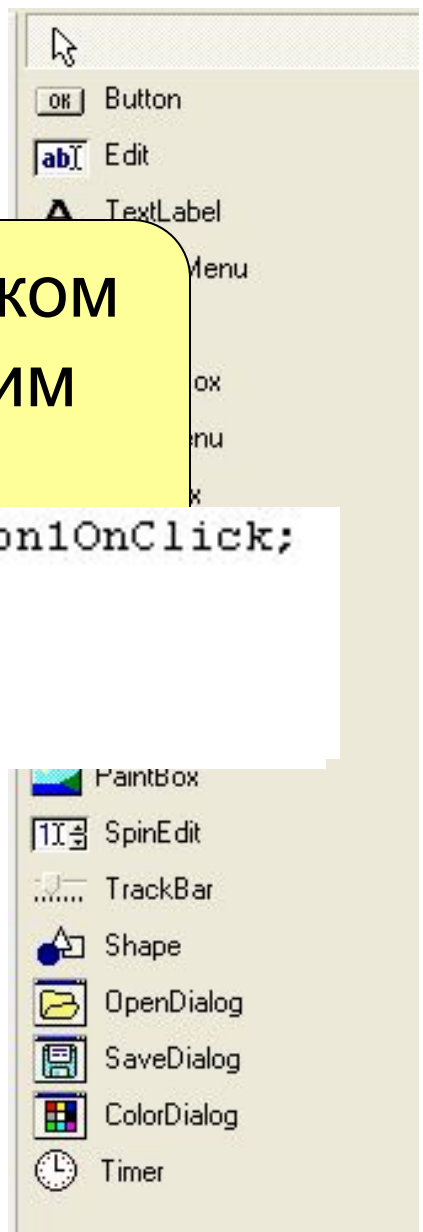


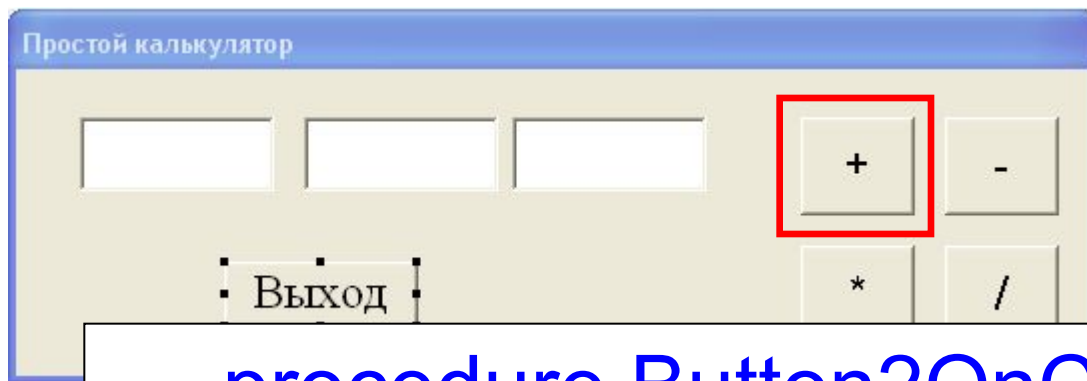
Двойным щелчком  
мыши создадим  
событие

```
procedure Button1OnClick;  
begin  
    form1.close;  
end;
```



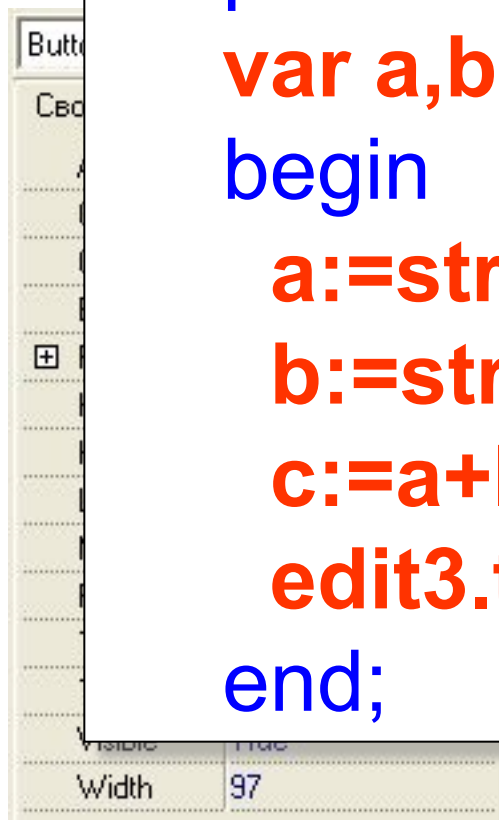
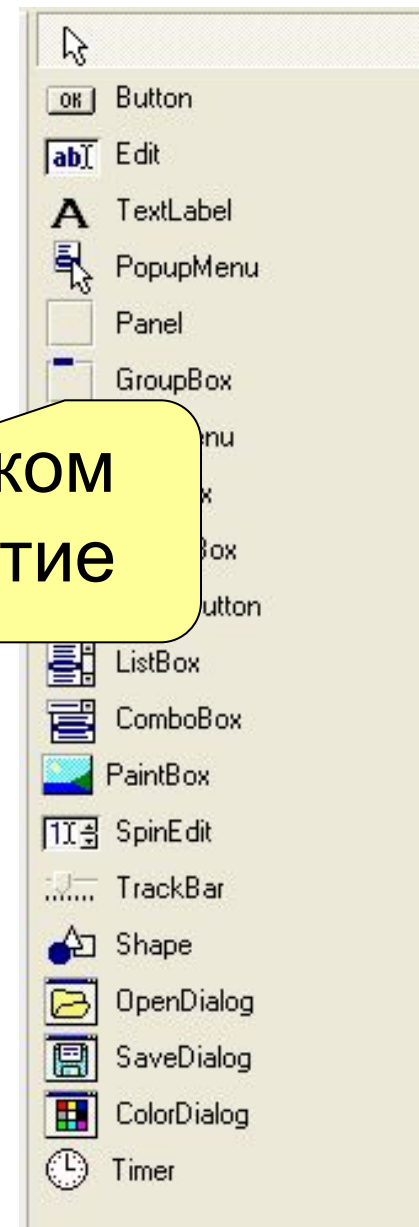
Добавим строку  
закрывающую форму  
**Form1.close;**

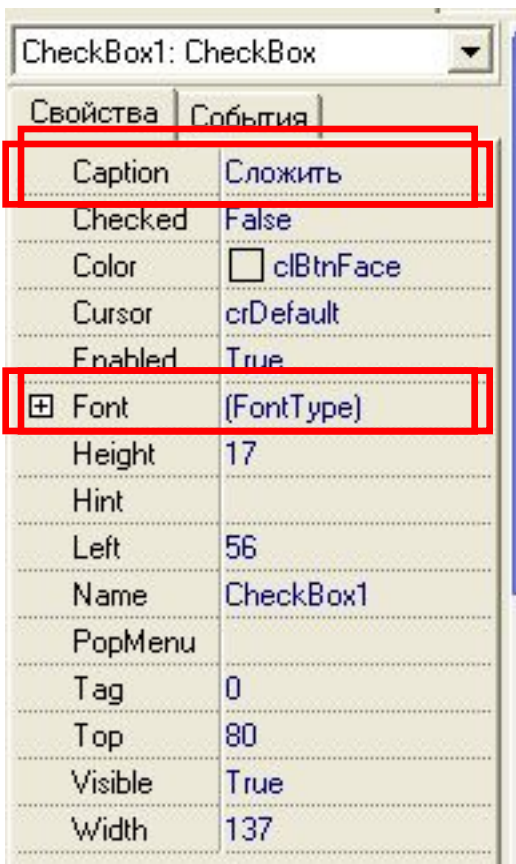
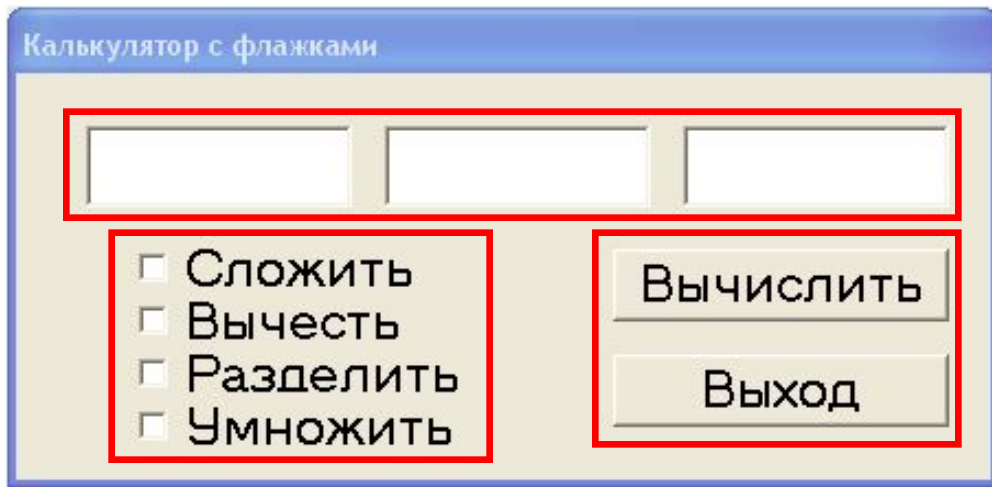




```
procedure Button2OnClick;  
var a,b,c:real;  
begin  
  a:=strToFloat(edit1.text);  
  b:=strToFloat(edit2.text);  
  c:=a+b;  
  edit3.text:=FloatToStr(c);  
end;
```

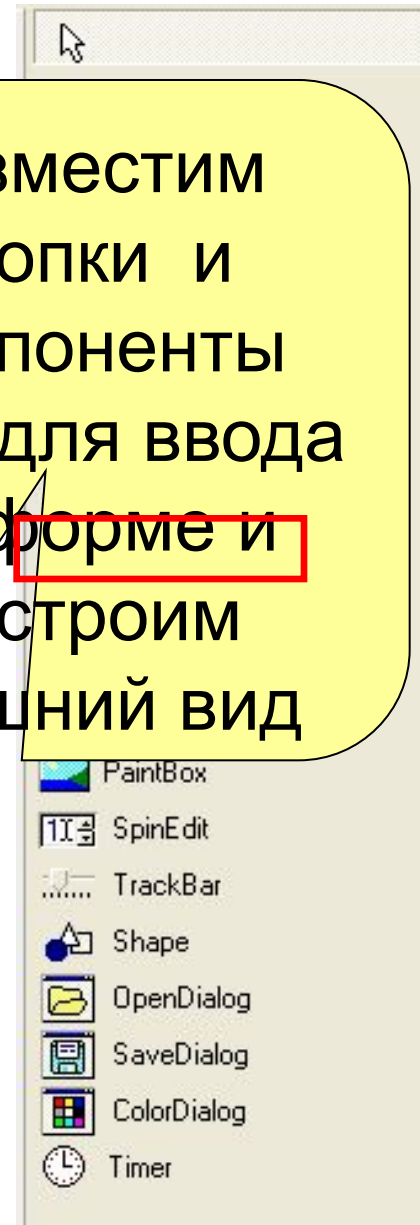
ТЧКОМ  
ЫТИЕ








Настроим свойства флажок (CheckBox)

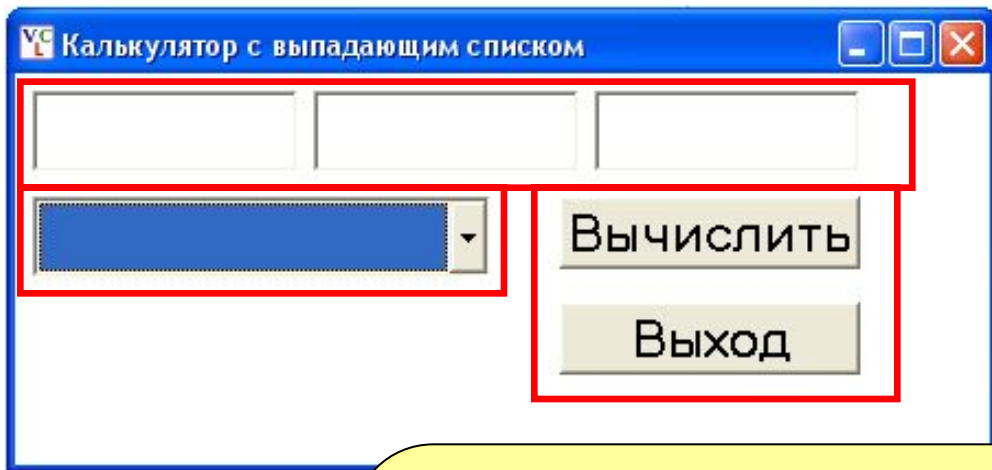
Разместим кнопки и компоненты поле для ввода на форме и настроим внешний вид



```
procedure Button1OnClick;  
  var a,b,c:real;  
begin  
  a:=strToFloat(edit1.text);  
  b:=strToFloat(edit2.text);  
  if (Checkbox1.Checked=true) then c:=a+b;  
  if (Checkbox2.Checked=true) then c:=a-b;  
  if (Checkbox3.Checked=true) then c:=a/b;  
  if (Checkbox4.Checked=true) then c:=a*b;  
  edit3.text:=floatToStr(c);  
end;
```

Name	CheckBox1
PopupMenu	
Tag	0
Top	80
Visible	True
Width	137

	SaveDialog
	ColorDialog
	Timer



Разместим на форме компонент выпадающий список (ComboBox)

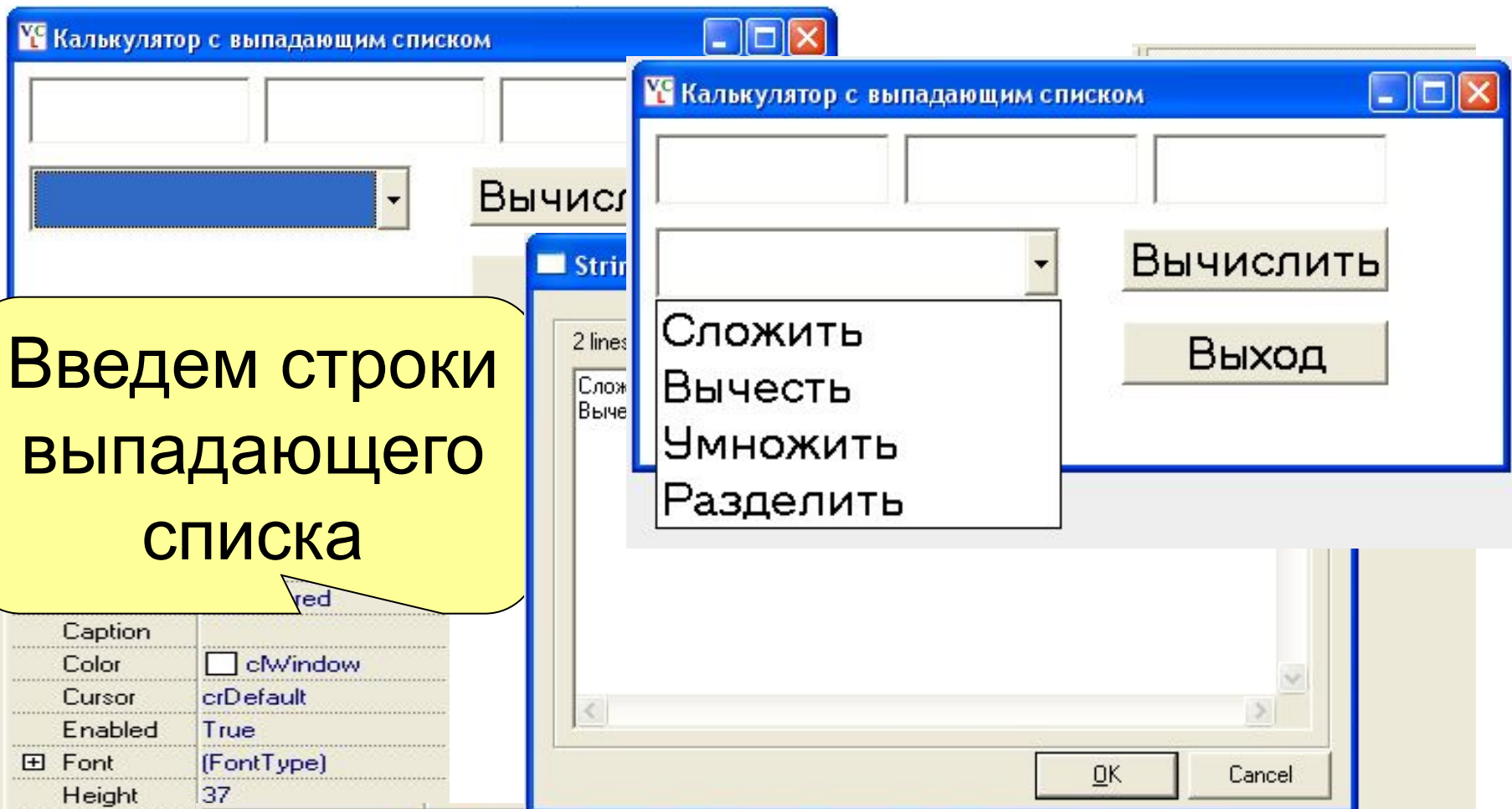
Разместим кнопки и компоненты поле для ввода на форме и настроим внешний вид

ComboBox1: ComboBox	
Свойства	События
Align	alNone
BevelInner	bvRaise
BevelOuter	bvLower
Caption	
Color	<input type="checkbox"/> clWhite
Cursor	crDefault
Enabled	True
Font	(FontType)
Height	37
Hint	
ItemIndex	0
Items	(StringArray)
Left	8
Name	ComboBox1
PopupMenu	
Tag	0
Top	56
Visible	True
Width	209

- PaintBox
- SpinEdit
- TrackBar
- Shape
- OpenDialog
- SaveDialog
- ColorDialog
- Timer



Введем строки выпадающего списка



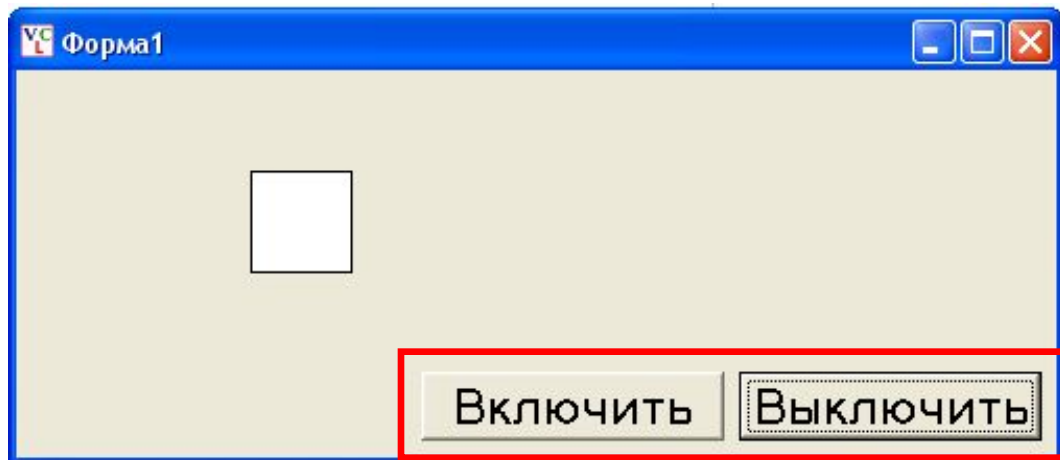
Caption	
Color	<input type="checkbox"/> clWindow
Cursor	crDefault
Enabled	True
Font	(FontType)
Height	37
Hint	
ItemIndex	
Items	[StringArray]
Left	8
Name	ComboBox1
PopupMenu	
Tag	0
Top	56
Visible	True
Width	209

Items [StringArray] ...

- Shape
- OpenFileDialog
- SaveDialog
- ColorDialog
- Timer

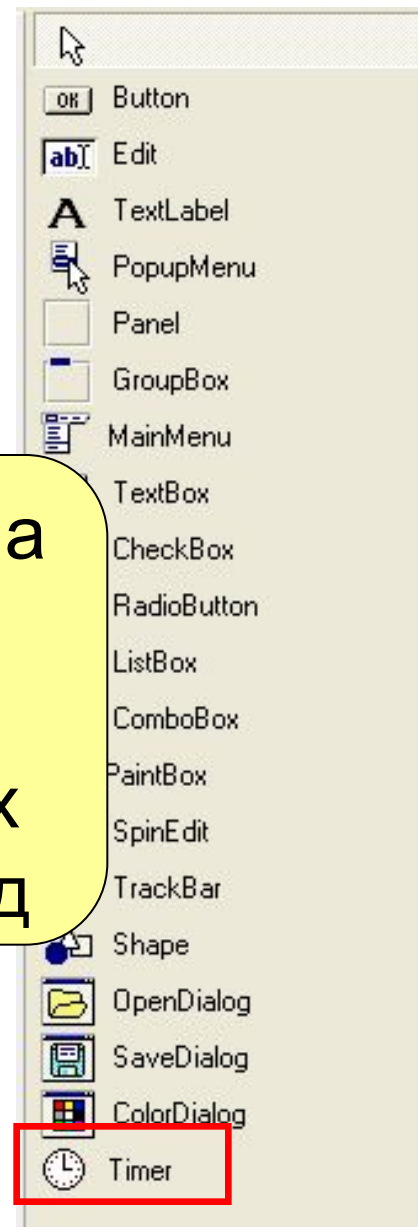
```
procedure Button1OnClick;  
var a,b,c:real;  
    k:integer;  
begin  
    a:=strToFloat(edit1.text);  
    b:=strToFloat(edit2.text);  
    k:=ComboBox1.ItemIndex;  
    if (k=1) then c:=a+b;  
    if (k=2) then c:=a-b;  
    if (k=3) then c:=a*b;  
    if (k=4) then c:=a/b;  
    edit3.text:=floatToStr(c);  
end;
```

Top	56
Visible	True
Width	209



Button1: Button

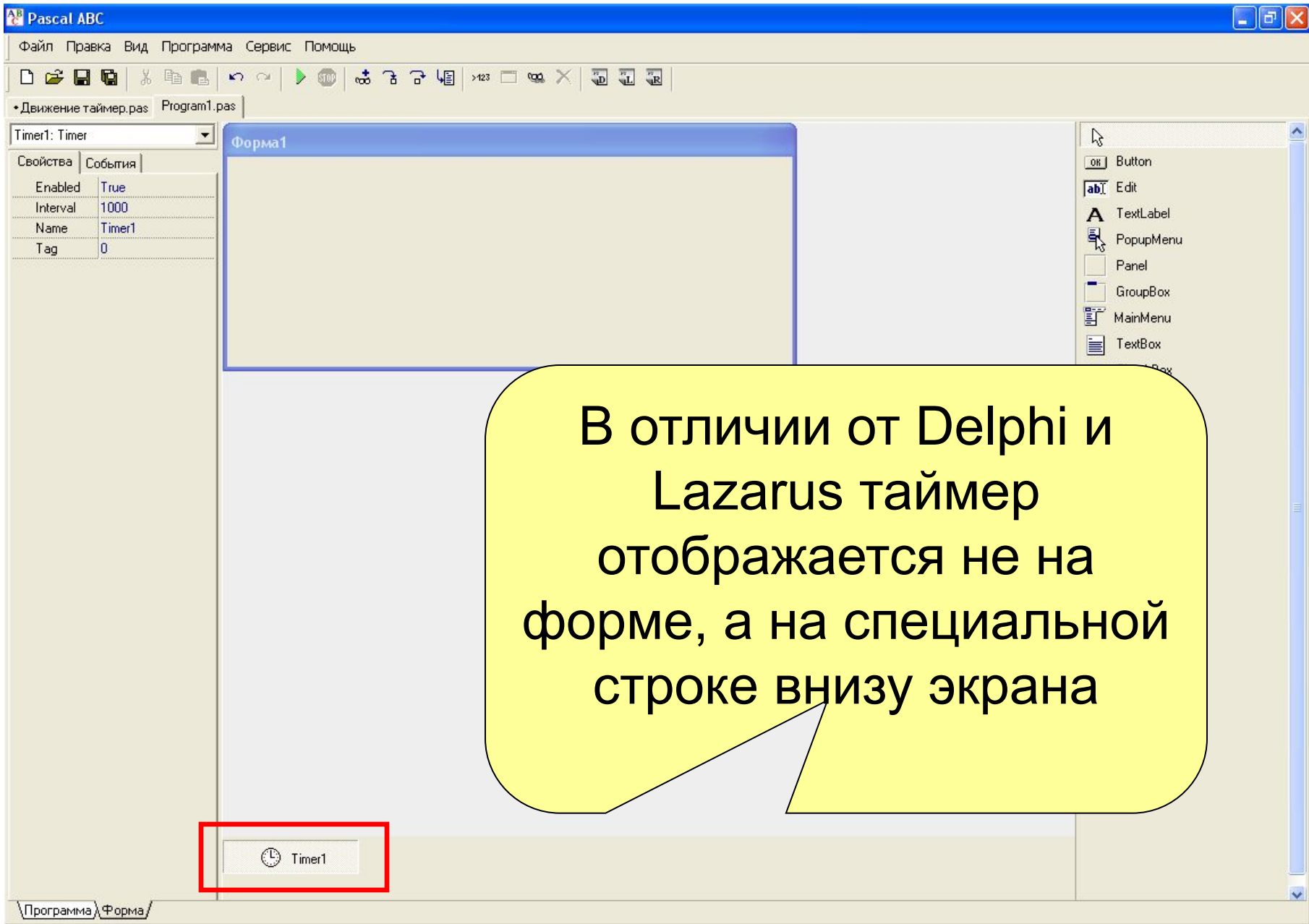
Свойства		События	
Align	alNone		
Caption	Включить		
Cursor	crDefault		
Enabled	True		
Font	(FontType)		
Height	33		
Hint			
Left	194		
Name	Button1		
PopupMenu			
Tag	0		
Top	144		
Visible	True		
Width	145		

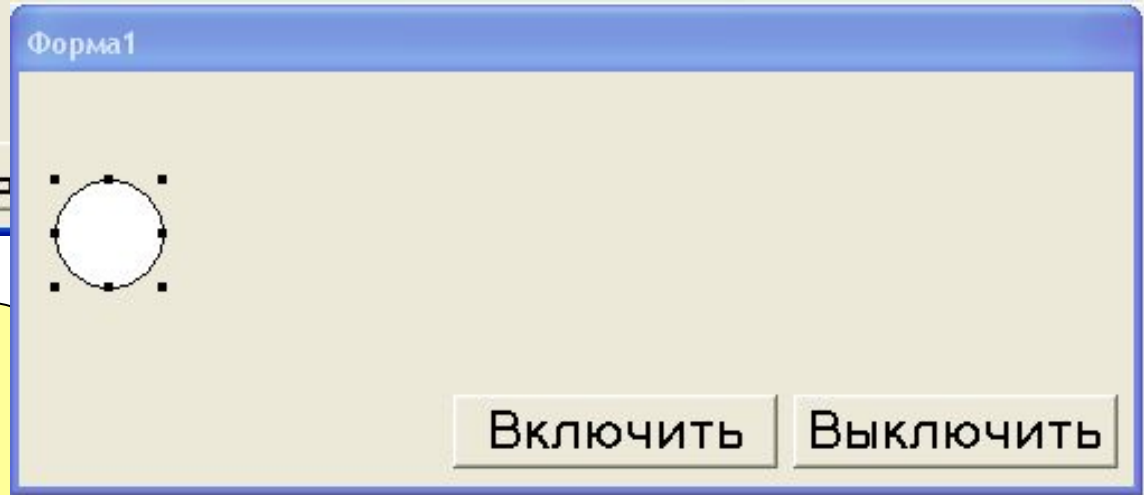
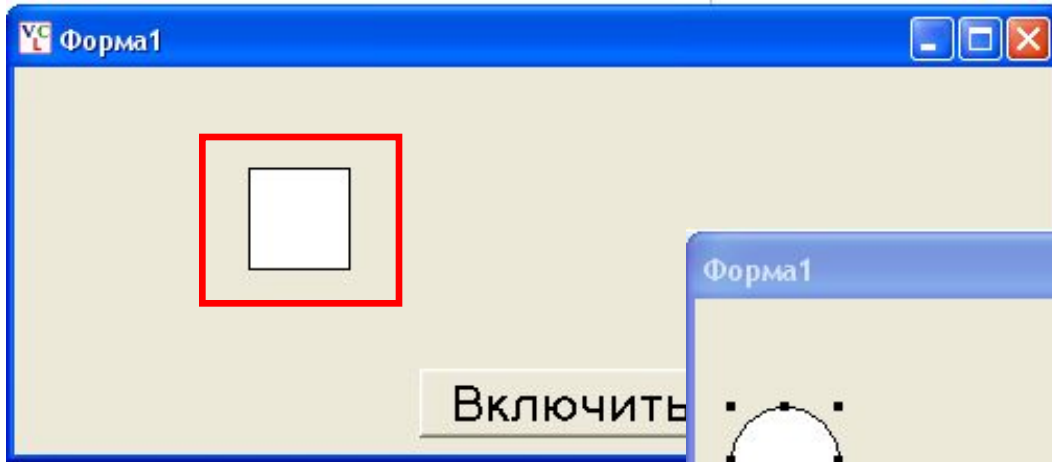


Разместим на форме две кнопки и

Разместим на форме таймер (Timer)

и оформим их следующий вид



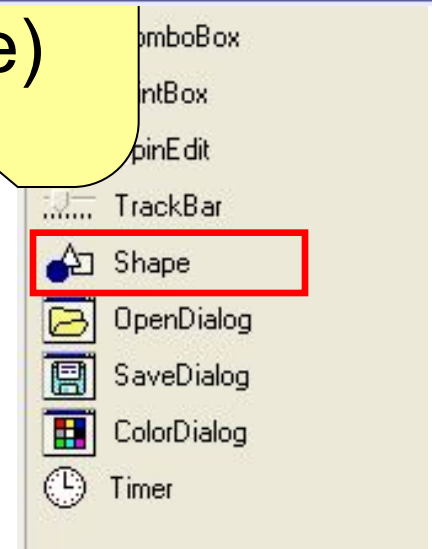


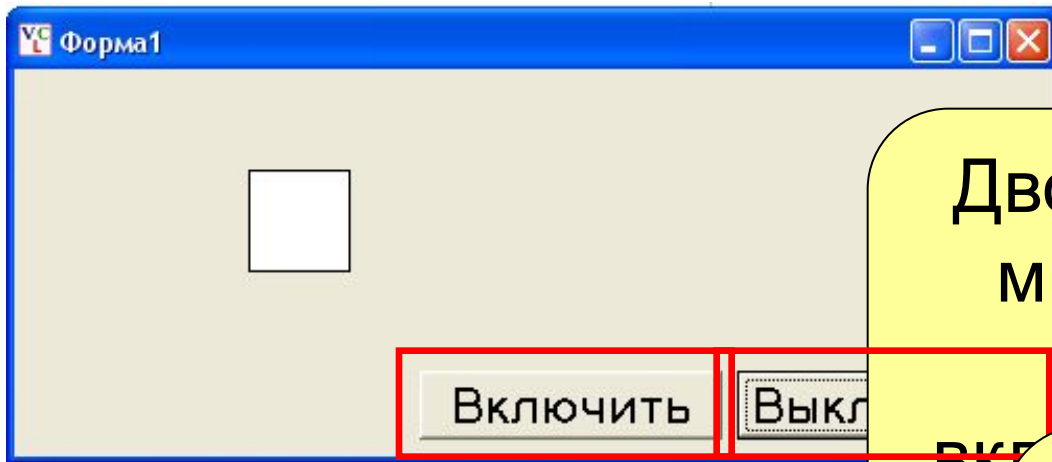
СВОЙСТВО В  
КОТОРОМ  
устанавливается  
вид фигуры

Фигура (Shape)

Left	184
Name	Shape1
PopupMenu	
Shape	stEllipse
Tag	0
Top	64
Visible	True
Width	65

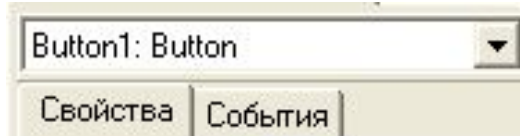
Shape	stEllipse
Tag	stRectangle
Top	stSquare
Visible	stRoundRect
Width	stRoundSquare
	stEllipse
	stCircle





Двойным щелчком мыши создадим событие для ВКЛ

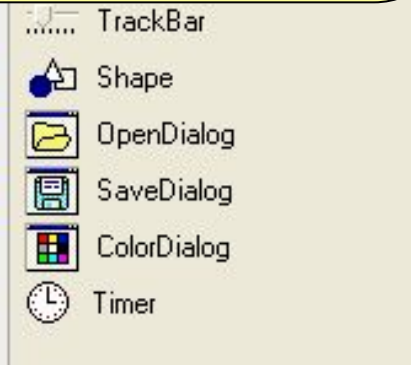
Двойным щелчком мыши создадим событие для выключения таймера



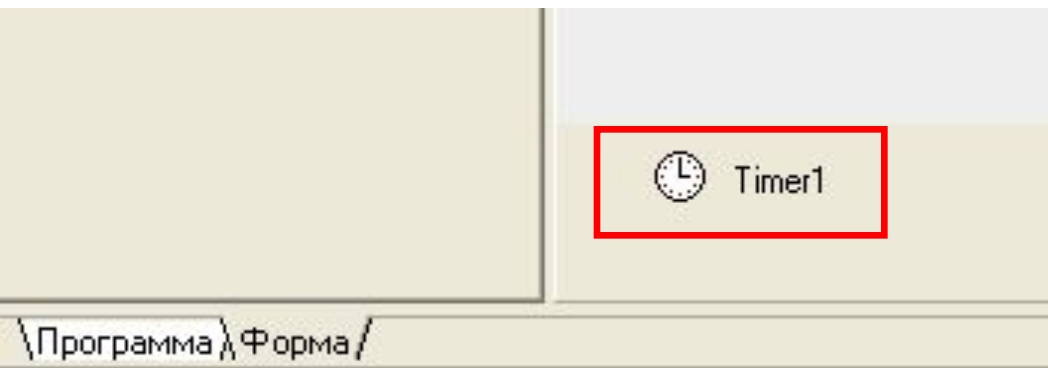
```

procedure Button1OnClick;
begin
    Timer1.Enabled:=False ;
end;
    
```

Tag	0
Top	144
Visible	True
Width	145



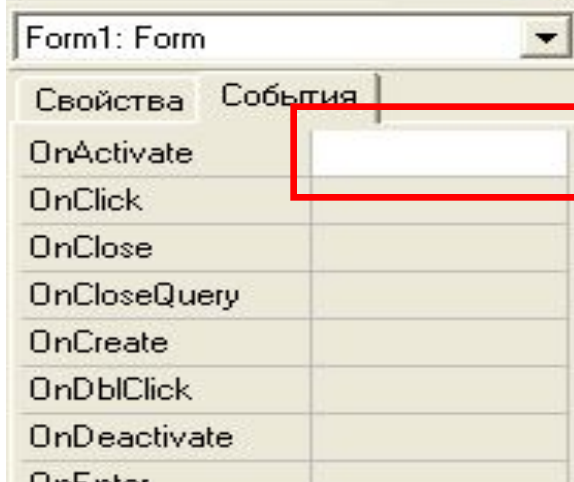
```
procedure Timer1OnTimer;  
begin  
    Shape1.Left:=Shape1.Left+d;  
    if (Shape1.Left>450) or (Shape1.Left<0) then d:=-d;  
end;
```



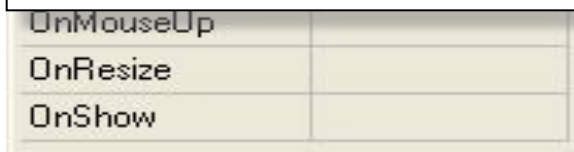
Двойным щелчком  
мыши создадим  
событие которое  
будет происходить  
при включенном  
таймере



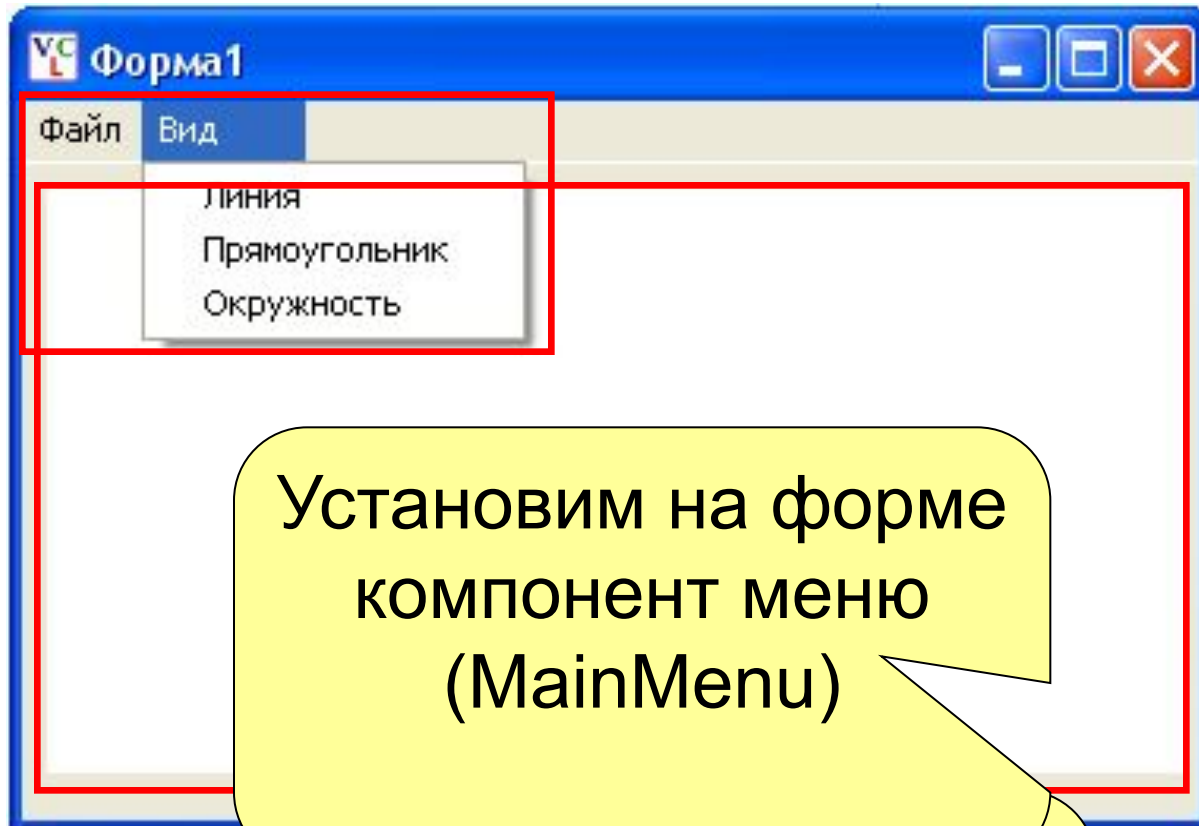
Двойным щелчком  
мыши по пустому полю  
создадим событие  
которое будет  
происходить при каждом  
запуске программы  
дурь  
которое будет происходить



```
procedure Form1OnActivate;  
begin  
    d:=1;  
end;
```

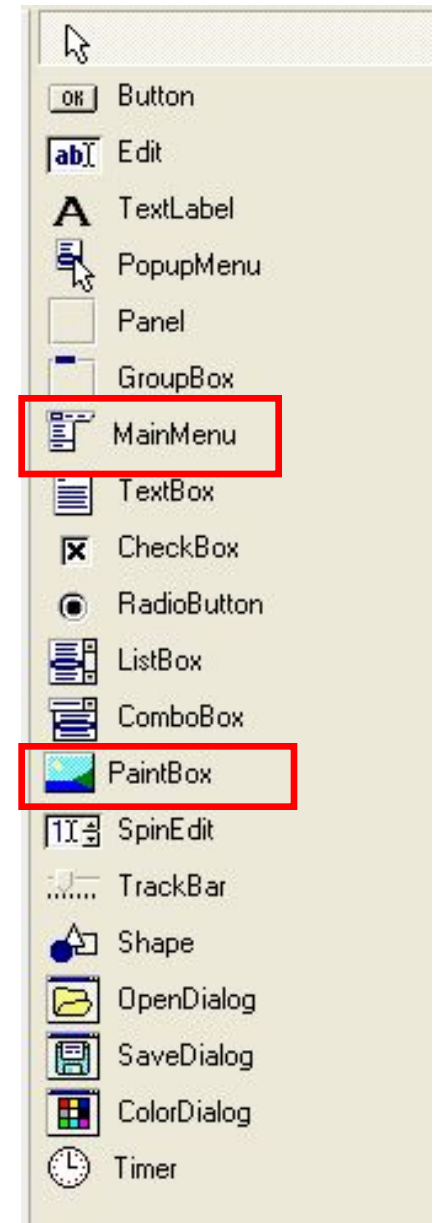


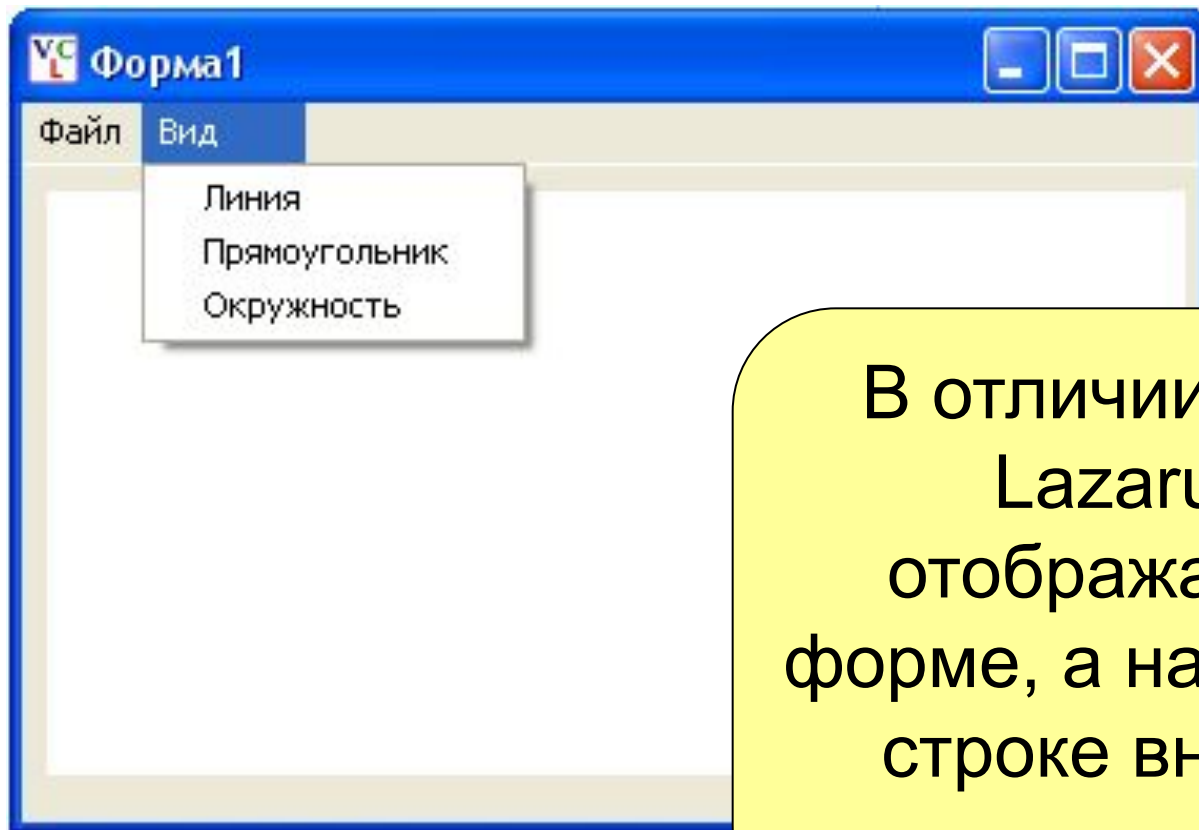




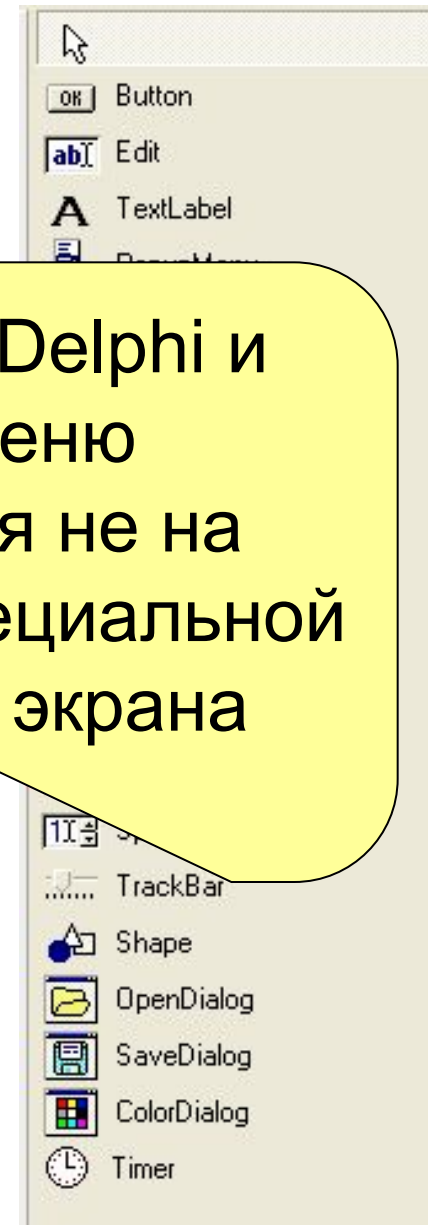
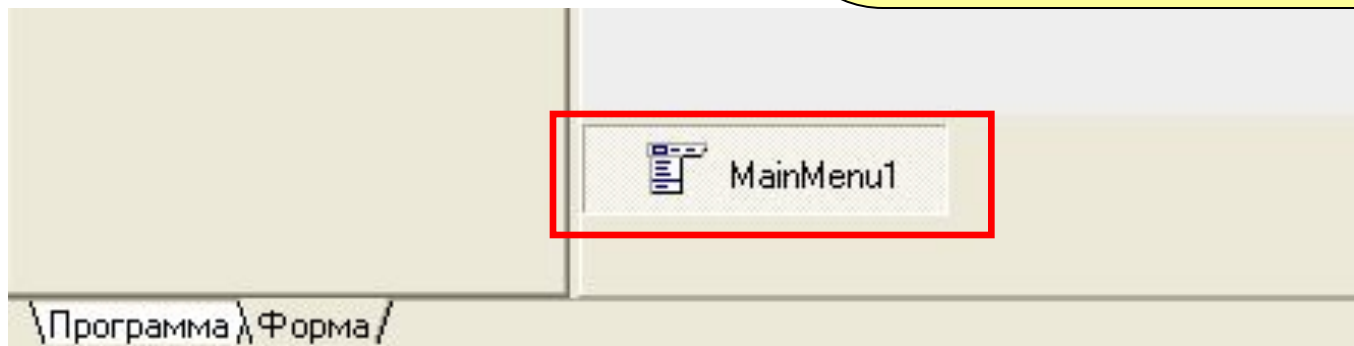
Установим на форме  
компонент меню  
(MainMenu)

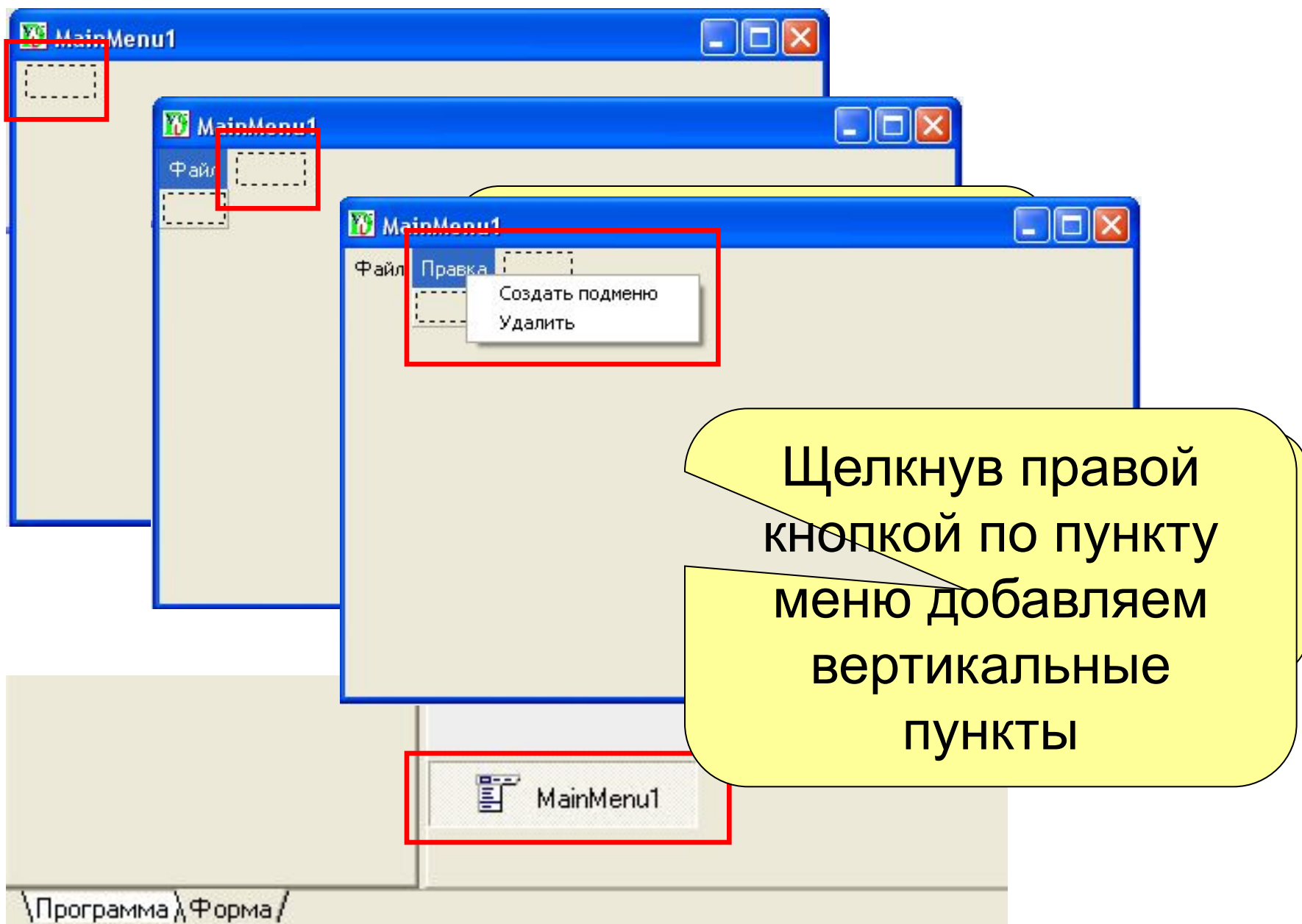
установим на форме  
панель для  
рисования (PictureBox)

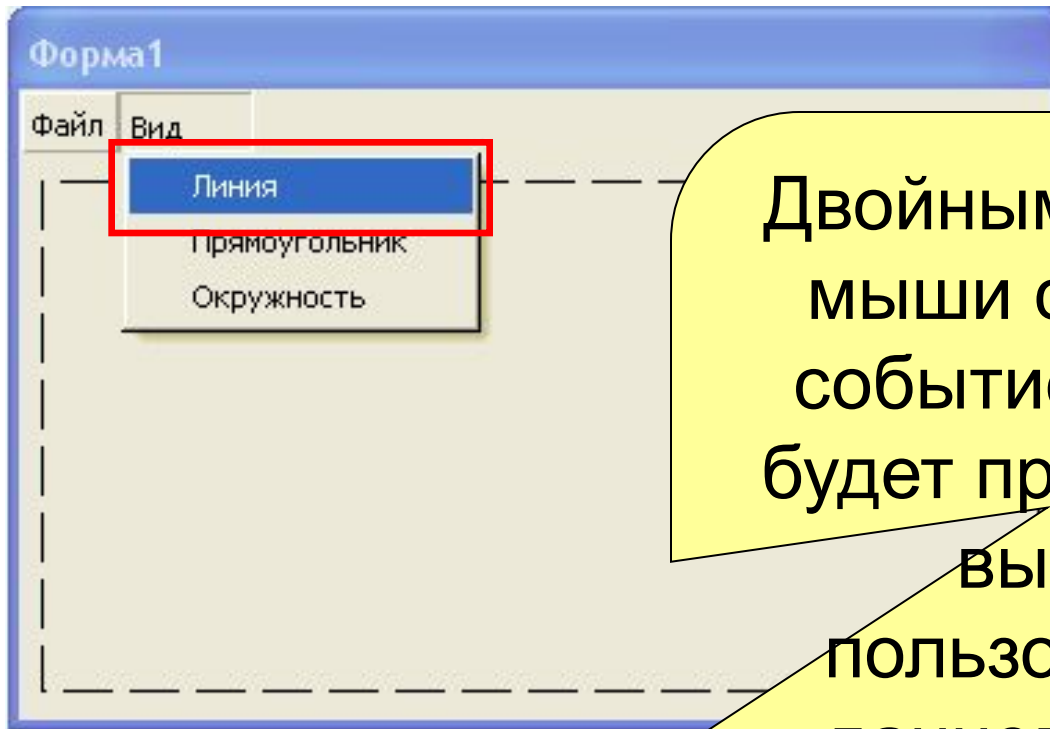




В отличии от Delphi и Lazarus меню отображается не на форме, а на специальной строке внизу экрана







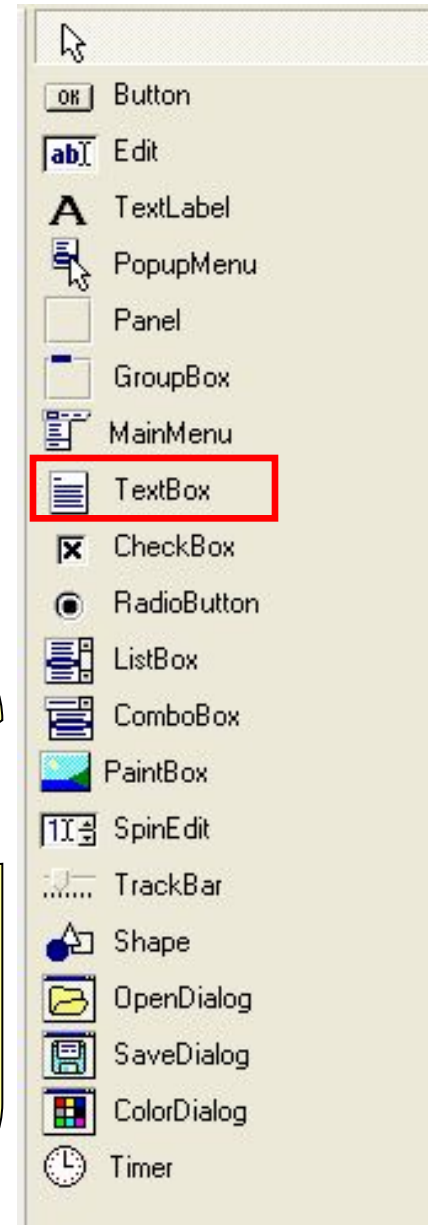
Двойным щелчком  
мыши создадим  
событие которое  
будет происходить  
выборе  
пользователем  
данного пункта

```
procedure MainMenu1_2_1OnClick;  
begin  
    PaintBox1.Canvas.Clear;  
    PaintBox1.Canvas.Line(0,0,350,100);  
end;
```



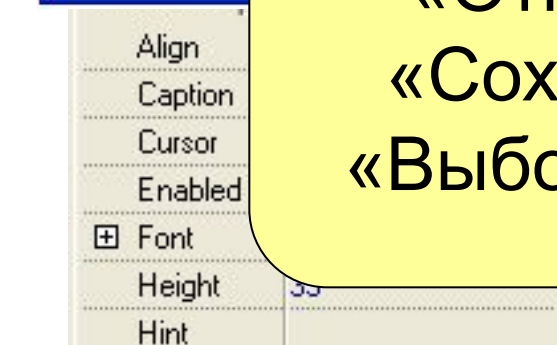
Align	alNone
Caption	Выход
Cursor	crDefault
Enabled	True
Font	(FontType)
Height	33
Hint	
Left	104
Name	Button5
PopupMenu	
Tag	0
Top	96
Visible	True
Width	97

Разместим на форме компонент многострочный редактор (TextBox)

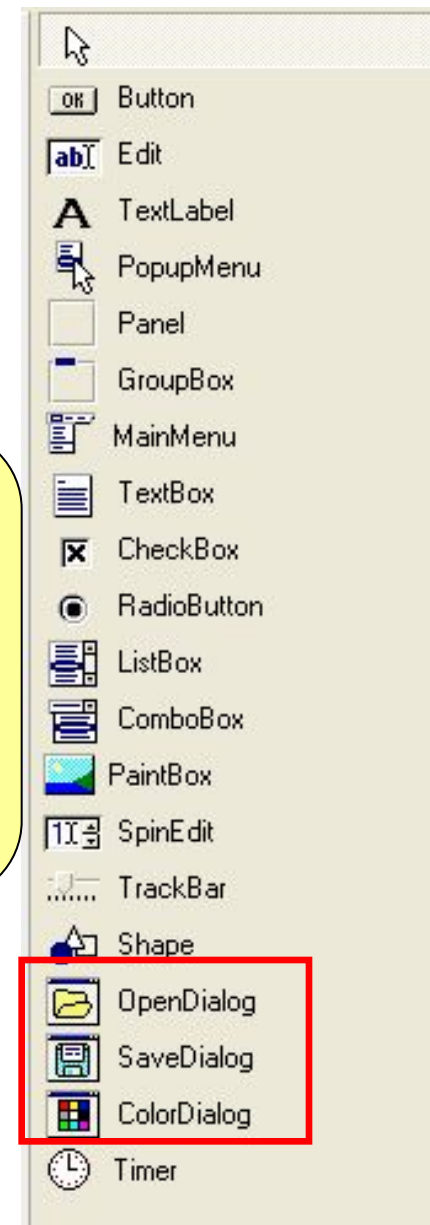




Разместим на форме диалоги  
«Открыть» (OpenDialog),  
«Сохранить» (SaveDialog)  
«Выбор цвета» (ColorDialog)



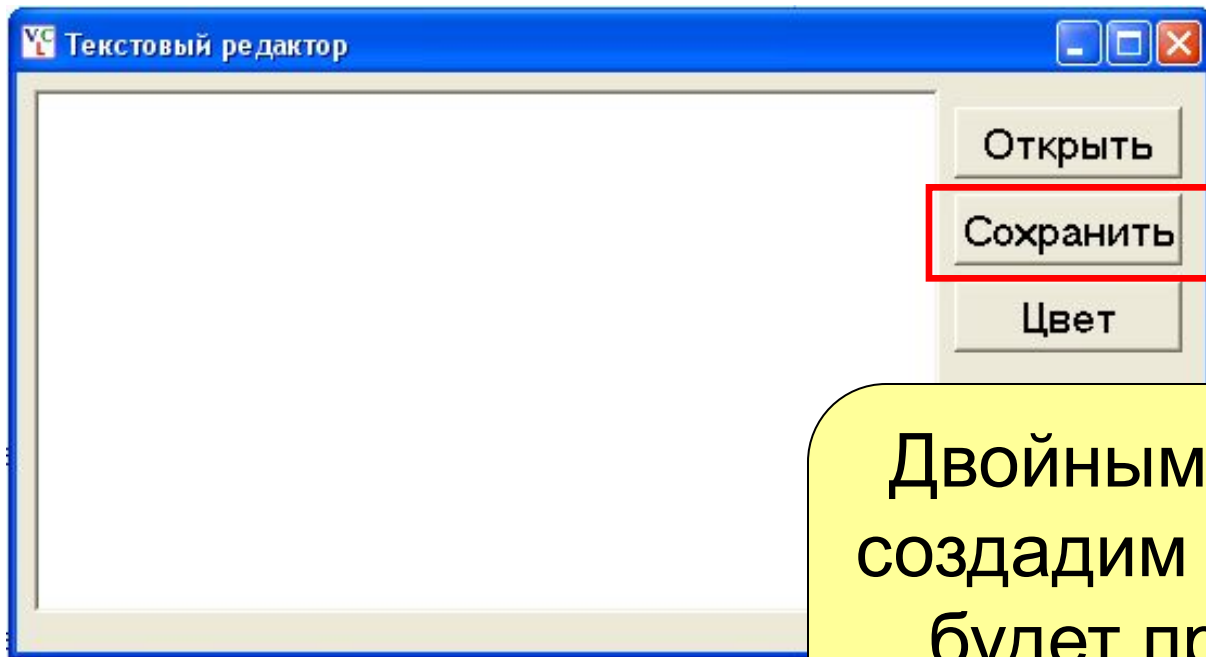
OpenDialog1 SaveDialog1 ColorDialog1





Двойным щелчком  
мыши создадим  
событие которое  
будет происходить  
при нажатии кнопки  
«Открыть»

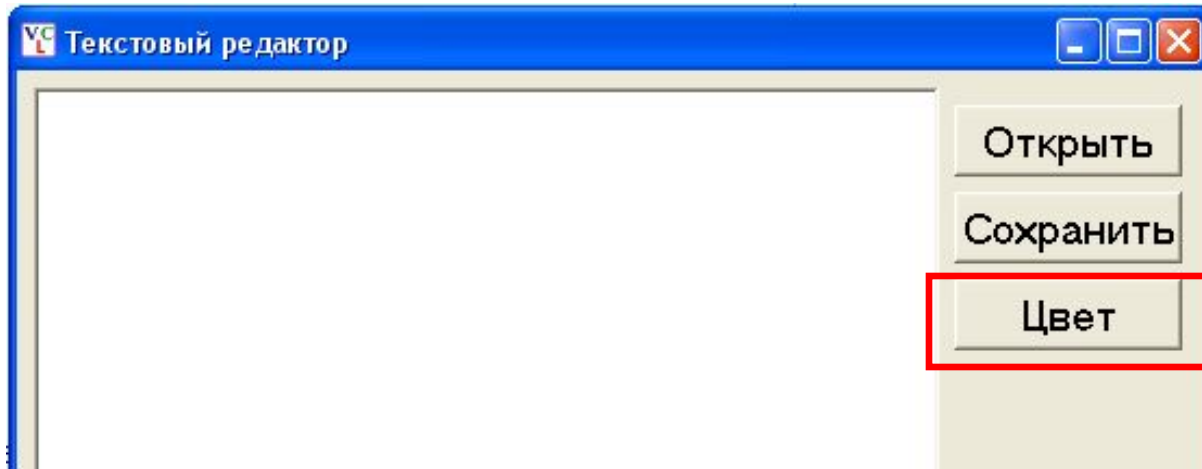
```
procedure Button1OnClick;  
begin  
if Opendialog1.Execute then  
TextBox1.Lines.LoadFromFile(Opendialog1.FileName);  
End;
```



Двойным щелчком мыши создадим событие которое будет происходить при нажатии кнопки «Сохранить»

```
procedure Button1OnClick;  
Begin  
  if SaveDialog1.Execute then  
    TextBox1.Lines.SaveToFile(SaveDialog1.FileName);  
End;
```





Двойным щелчком мыши создадим событие которое будет происходить при нажатии кнопки «Цвет»

```
procedure Button1OnClick;  
Begin  
    if ColorDialog1.Execute then  
        TextBox1.font.Color:= ColorDialog1.Color;  
End;
```