

# **Модульный принцип построения проекта и программного кода**

- Проект включает в себя программные модули форм и стандартные программные модули в виде отдельных файлов.

- Программный модуль формы может включать несколько процедур. В языке VB процедуры могут быть двух типов: **событийные** и **общие**. Событийные процедуры позволяют создавать интерактивные приложения, так как дают пользователю возможность выполнять тот или иной алгоритм с помощью определенного действия (например, щелчка по кнопке). Общая процедура начинает выполняться после её **вызова** из другой процедуры.

- Общая процедура вызывается на выполнение либо с помощью оператора **Call**, либо по имени.
- **Call** имяПроцедуры(Список параметров)  
иначе  
ИмяПроцедуры СписокПараметров

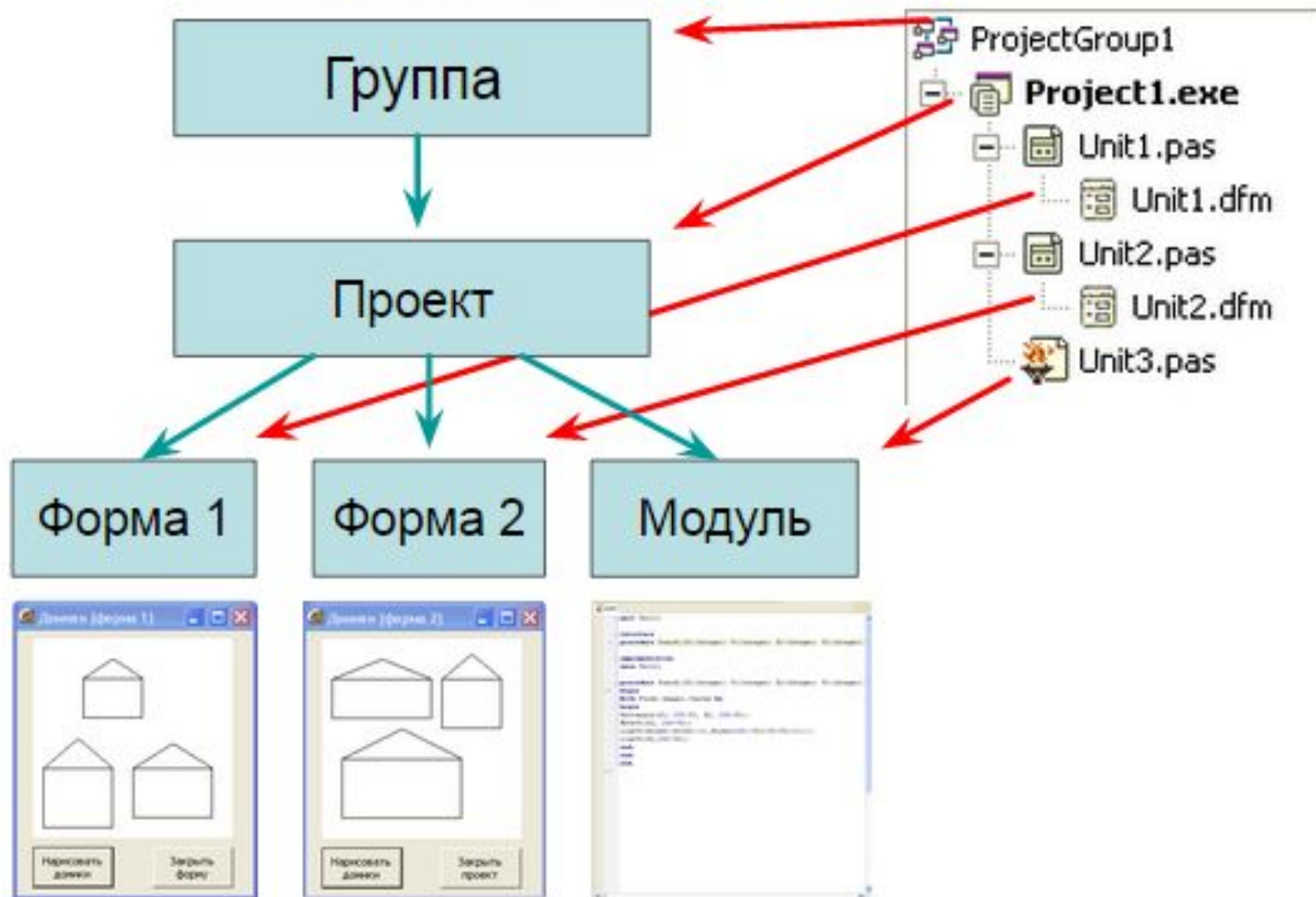
# Локальная процедура задаётся с помощью ключевого слова **Private:**

- **Private Sub** ИмяПроцедуры
- Программный код
- **End Sub**

Глобальная процедура задаётся  
с помощью ключевого слова  
**Public** (или по умолчанию, если  
ключевые слова перед **Sub**  
отсутствуют):

- **Public Sub** ИмяПроцедуры
- Программный код
- **End Sub**

# Проект «Домики»



## Процедура (подпрограмма) рисования домика

```
procedure Domik1(X1:Integer; Y1:Integer; X2:Integer; Y2:Integer);  
begin  
  With Form1.Image1.Canvas Do  
    begin  
      Rectangle(X1, 200-Y1, X2, 200-Y2);  
      MoveTo(X1, 200-Y2);  
      LineTo(Round((X1+X2)/2),Round(200-(Y2+(Y2-Y1)/2)));  
      LineTo(X2, 200-Y2);  
    end;  
end;
```

## Процедура-обработчик события рисования ДОМИКОВ

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
  Domik1(10, 10, 80, 70);  
  Domik1(100, 20, 180, 70);  
  Domik1(50, 120, 110, 160);  
end;
```





Процедура-обработчик события, в котором закрывается первая форма и открывается вторая

```
procedure TForm1.Button2Click(Sender: TObject);  
begin  
    Form1.Visible := false;  
    Form2.Visible := true;  
end;
```

