

Вам понадобился специфический компонент в Delphi, типа **TButton**, но который будет иметь больше свойств или методов, чем стандартная кнопка? Опытные разработчики Delphi, когда им нужен компонент

### **TSuperBtn**

, применяют два подхода к этой проблеме: ищут готовые сторонние VCL решения или же создают свои собственные компоненты.

Ну а если Вам не нужно, чтобы **TSuperBtn** находился на *Палитре Компонентов* и нужен только для этого приложения? Как получить

### **TButton**

с большим количеством свойств и методов, но не

### **TSuperBtn**

? И как расширить возможности

### **TButton**

без необходимости создавать унаследованный класс с другим именем?

Это можно сделать при помощи вспомогательного класса, но вспомогательный класс будет содержать только вновь добавленные свойства и методы.

Большинство начинающих не знают, что они *МОГУТ* создавать свои собственные пользовательские элементы управления на основе существующего набора VCL, создавая, так называемый класс-перехватчик, который будет иметь то же самое имя, что и основной класс.

## Перехватчик классов Delphi

Во-первых, два класса с одним именем можно применять в Delphi, если эти два класса находятся в разных модулях. Поэтому следующее объявление является правильным:

```
type TButton = class(StdCtrls.TButton)
```

Приведенный выше код создает перехватчик класса для управления **TButton**, оригинал

### **TButton**

объявлен в модуле

*StdCtrls*

библиотеки VCL.

К тому же это объявление должно быть помещено в отдельный модуль.

И самое главное: модуль перехватчика классов должен быть объявлен *ПОСЛЕ* перечисления модуля основного класса.

```
uses { ... } StdCtrls, Unit2 { ... } Пример перехватчика класса TButton =  
class(StdCtrls.TButton)
```

Вот один перехватчик класса для **TButton**.

Добавляем новое свойство **Text**, которое будет хранить строку, а также переопределим процедуру *Click*, которая будет менять надпись на кнопке, а также выполнять свою основную деятельность.

```
unit Unit2; interface uses StdCtrls, SysUtils; type TButton = class(StdCtrls.TButton)  
private fText: string; public procedure Click; override; property Text: string read  
fText write fText; end; implementation { TButton } procedure TButton.Click; begin  
Caption:= 'Это кнопка'; inherited; end; end.
```

Теперь поместите кнопку на форму и добавьте модуль *Unit2* в конце перечисления модулей в разделе **uses**, главное, чтобы это было после **StdCtrls**.

В форме для примера у меня такой код:

```
procedure TForm1.Button1Click(Sender: TObject); begin Button1.Text:= 'Hello';  
Caption:= Button1.Text; end;
```

При выполнении программы и нажатии на кнопку, кнопка поменяет свою надпись, а форма – заголовок.

Вариантов использования перехватчика классов может быть много, в зависимости от

Ваших потребностей и возможностей.

Ну, пожалуй, это все, основные моменты рассмотрены. Если кто и не знал о перехвате классов, то начало положено.

Также заметьте, что это может быть применено не только к компонентам VCL, но и к любым классам Delphi.