



Предполагается, что Вы добавили данные в формате **RCDATA**. Это определенный пользователем тип данных. Для чтения ресурсов очень просто использовать класс **TResourceStream**

, который поставляется вместе с Delphi.

TResourceStream

можно создать таким способом:

```
var RS: TResourceStream; begin // Do this if the resource is named
RS := TResourceStream.Create( HInstance, // хендл приложения, которое содержит DLL
ResourceName, // строка с именем ресурса RT_RCDATA); // идентификатор типа
RCDATA RS := TResourceStream.CreateFromID( HInstance, ResourceID
RT_RCDATA); end;
```

Давайте создадим текстовый файл, например, *MYRES.RC*. В нем пишем следующее:

```
TEXTRTF RCDATA text.rtf
```

В этом примере в качестве данных будем использовать текстовый документ формата **RTF**

Файл

RFT

с каким-либо текстом лежит в той же папке, что и *MYRES.RC*

Далее, помещаем файл **BRCC32.EXE** в эту же папку. Пишем файл *Compile.bat* со следующим содержанием:

```
brcc32.exe myres.rc myres.res
```

После его запуска, получаем файл *MYRES.RES* в этой же папке.

Подключаем файл ресурсов к нашему проекту:

```
var Form1: TForm1; implementation {$R *.dfm} {$R MYRES.RES}
```

Помещаем на форму компонент **TRichEdit** (*RichEdit1*) и кнопку **TButton** (*Button1*).

Пишем следующий код:

```
unit Unit1; interface uses Windows, Messages, SysUtils, Variants, Classes,
Graphics, Controls, Forms, Dialogs, StdCtrls, ComCtrls; type TForm1 = class(TForm)
Button1: TButton; RichEdit1: TRichEdit; procedure Button1Click(Sender: TObject);
private { Private declarations } public { Public declarations } end; var Form1:
TForm1; implementation {$R *.dfm} {$R MYRES.RES} procedure
TForm1.Button1Click(Sender: TObject); var RS: TResourceStream; begin // Создаем
поток ресурсов (идентификатор 100) RS := TResourceStream.CreateFromID(HInstance,
100, RT_RCDATA); try // Загружаем данные в компонент RichEdit
RichEdit1.Lines.LoadFromStream(RS); finally // Освобождаем память RS.Free; end;
end; end.
```