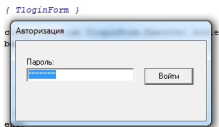


MainForm - это форма (окно), которая создается первой при запуске приложения Delphi. Если Вам нужно вывести диалоговое окно ввода пароля или логина, его нужно создать и вывести перед созданием основной формы.

MainForm



Когда создается новый проект Delphi, автоматически создается *Form1* и придает ей свойства главной формы -

MainForm

(глобального объекта

Application

). Чтобы назначить другой форме свойство

MainForm

, откройте вкладку

Forms

диалогового окна Project -> Options и в списке

MainForm

выберите нужную форму.

Когда главная форма закрывается, приложение завершает свою работу.

Диалоговое окно Авторизация

Начнем с создания главной формы приложения. Создайте новый проект, который содержит одну форму. Это будет главная форма проекта.

Измените имя формы на *mainForm* и сохраните этот модуль под именем *main.pas*, а сам проект под именем *Password.dpr*.

Теперь добавьте вторую форму в проект и дайте ей имя *loginForm*. По умолчанию, вторая форма тоже будет создаваться автоматически. Удалите ее из авто-создания формы, перейдя в меню Project -> Options на вкладке

Forms

. Сохраните этот модуль под именем `login.pas`.

Код проекта

```
program Password; uses Forms, login in 'login.pas' {loginForm}, main in 'main.pas'
{mainForm}; {$R *.res} begin Application.Initialize; Application.CreateForm(TmainForm,
mainForm); Application.Run; end.
```

Добавьте на форму **TButton** (`btnLogin`), **TEdit** (`edPassword`) и **TLabel** (`lblPassword`).

Добавьте метод класса создания и уничтожения формы диалогового окна авторизации.

Функция **Execute** возвращает *True*, если пользователь ввел правильный пароль в поле ввода пароля.

Вот исходный код формы авторизации:

```
unit login; interface uses Windows, Messages, SysUtils, Variants, Classes, Graphics,
Controls, Forms, Dialogs, StdCtrls; type TloginForm = class(TForm) edPassword:
TEdit; btnLogin: TButton; lblPassword: TLabel; procedure btnLoginClick(Sender:
TObject); private { Private declarations } public { Public declarations } class
function Execute: boolean; end; var loginForm: TloginForm; implementation {$R *.dfm}
{ TloginForm } class function TloginForm.Execute: boolean; begin with
TloginForm.Create(nil) do try result:= ShowModal = mrOk; finally Free; end; end;
procedure TloginForm.btnLoginClick(Sender: TObject); begin if edPassword.Text =
'password' then ModalResult:= mrOk else ModalResult:= mrAbort; end; end.
```

Метод **Execute** динамически создает экземпляр **TloginForm** и отображает его модально при помощи метода **ShowModal**.

Программа из **ShowModal** не возвращается, пока форма не будет закрыта. Когда форма закрывается, она возвращает значение свойства **ModalResult**.

Обработчик события **OnClick** кнопки `btnPassword` присваивает свойству **ModalResult** *mrOk*,

если пользователь ввел правильный пароль или

mrAbort

, если пользователь ввел неверный пароль. Таким образом, главная форма не будет создаваться, если пользователь ввел неверный пароль.

Вот исходный код проекта с условиями перехода:

```
program Password; uses Forms, login in 'login.pas' {loginForm}, main in 'main.pas'
{mainForm}; {$R *.res} begin if TloginForm.Execute then begin Application.Initialize;
Application.CreateForm(TmainForm, mainForm); Application.Run; end else begin
Application.MessageBox('У Вас нет прав на использование этой ' + 'программы.
Пароль "password", 'Защита паролем'); end; end.
```

Вот и все!

[Скачать проект](#)