



Томский государственный университет



Факультет информатики

А.В. Скворцов, Т.Н. Поддубная

Меню и диалоги

**(методические указания к
лабораторной работе № 5)**

Томск – 1999

Указания РАССМОТРЕНЫ и УТВЕРЖДЕНЫ методической комиссией факультета информатики.

Протокол № _____ от « _____ » _____ 1999 г.

Председатель методической комиссии факультета информатики,
профессор _____ Поддубный В.В.

Утверждено. Зав. кафедрой прикладной информатики, доцент
_____ Сущенко С.П. « _____ » _____ 1999 г.

Методические указания посвящены программной среде Borland Delphi компании Inprise Corporation – одному из мощнейших современных средств разработки приложений для Windows.

Указания разработаны для студентов межфакультетской специализации факультета информатики Томского государственного университета. Содержат подробное описание лабораторной работы: цель, задачи, описание работы и используемых компонентов Delphi. В конце работы приведены тексты программ.

Рецензент – канд. техн. наук, доцент **Ю.Л. Костюк.**

© Скворцов А.В., Поддубная Т.Н., 1999

© Оформление и верстка: Скворцов А.В., 1999

В данной лабораторной работе вы познакомитесь с компонентами Delphi для создания меню и вызова стандартных диалогов выбора файлов. При этом необходимо будет создать приложение, позволяющее просматривать и запускать приложения, которые были созданы во время выполнения предыдущих лабораторных работ.

Управляющим элементом создаваемого приложения должно быть меню, позволяющее выбирать требуемые для просмотра или запуска файлы. Если выбрана команда просмотра, то содержимое файла должно быть выведено в главной форме в компоненте для отображения многострочных текстов. При выборе команды запуска соответствующий выполняемый файл должен быть запущен.

Цель работы

Изучение компонентов для создания меню и стандартных диалогов.

Задачи работы

1. Ознакомление с компонентами для создания меню и диалогов выбора файлов.
2. Создание программы для просмотра и запуска созданных в предыдущих лабораторных работах приложений.

Описание работы

Наше приложение будет состоять всего из одной формы, на которой надо разместить 4 компонента типов TMainMenu, TMemo и две TOpenDialog (рис. 16). С помощью этих компонентов мы реализуем два способа решения поставленной задачи. Рассмотрим их.

Для создания главного меню в Delphi используется компонент типа TMainMenu. Для этого необходимо поместить его на форму и дважды щелкнуть по нему. После этого на экране появится окно редактора меню (рис. 17).

Сначала меню состоит из одного пустого элемента. Для добавления нового пункта меню можно просто начать изме-

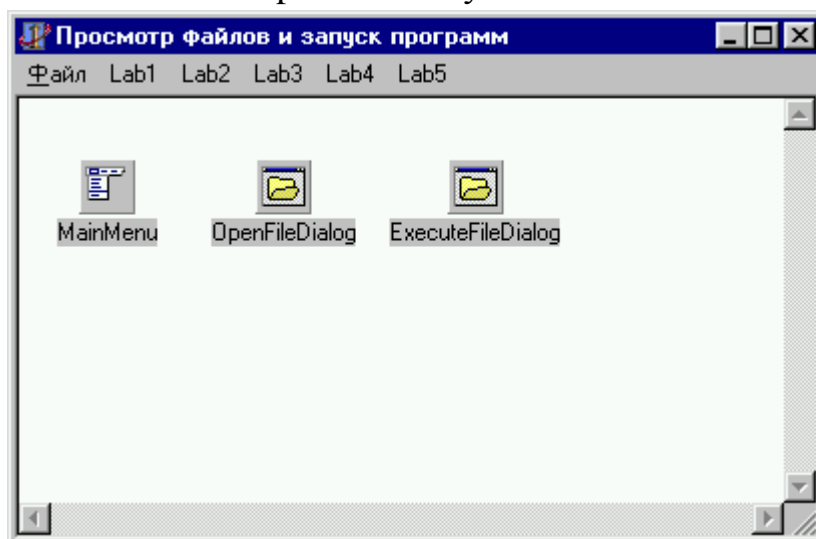


Рис. 16. Главная форма приложения

нять любой пустой пункт меню в редакторе меню (он обведен пунктирным прямоугольником). Для вставки нового пункта меню в любой позиции необходимо нажать клавишу Insert. Для удаления пункта меню – клавишу Delete. Для создания подменю второго уровня – клавишу Ctrl+→.

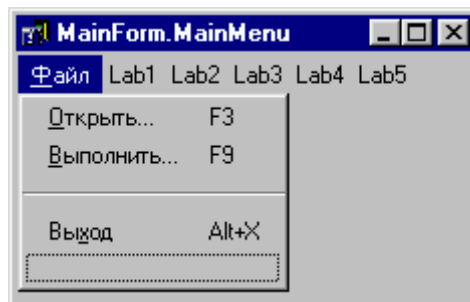


Рис. 17. Редактор меню

Для изменения описания пунктов меню необходимо выбрать требуемый. При этом он станет доступным в инспекторе объектов, где можно изменить все необходимые свойства. В табл. 8 приведено краткое описание основных свойств элементов меню.

Замечание

Для изменения пунктов меню используйте клавиши Insert, Delete и Ctrl+→.

Таблица 8. Основные свойства объектов типа TMenuItem

Свойство	Тип	Комментарий
Caption	String	Текст, появляющийся в меню. Если перед некоторой буквой поставить знак &, то во время выполнения программы этот знак будет не виден, а буква будет изображаться с подчеркиванием, что позволит ее использовать в качестве горячей клавиши. Если текст состоит всего из одного знака «минус», то пункт меню становится разделительной линией.
Checked	Boolean	При установке значения True слева от пункта меню будет отображаться «галочка».
Enabled	Boolean	Если значение будет равно False, то пункт меню будет недоступен для выбора и станет отображаться серым цветом.
ShortCut	TShortcut	Горячая клавиша, которую можно будет использовать для быстрого вызова пункта меню. Кроме того, она будет отображаться справа от пункта меню.
Visible	boolean	Если значение будет равно False, то пункт меню будет невидим

В данной лабораторной работе необходимо создать в главном меню два вида подменю. В первом подменю «Файл» можно создать три пункта и один разделитель. Это команды «Открыть...» для выбора файла в диалоге и загрузки его в главную форму, «Выполнить...» для выбора выполняемого файла в диалоге и его запуска, а также команда «Выход».

Все остальные подменю должны соответствовать предыдущим лабораторным работам и оформлены примерно так, как показано на рис. 18. Первым пунктом такого подменю должно быть название файла проекта, например Lab1.dpr. Затем после разделителя список всех модулей, входящих в состав проекта. А в конце после разделителя имя выполняемого файла.

После создания меню необходимо создать обработчики событий выбора пунктов меню. Для их создания можно просто дважды щелкнуть в нужный пункт меню, находясь в редакторе меню, либо просто выбрать этот пункт в меню формы.

Самый простой из обработчиков соответствует пункту меню «Выход». В нем просто нужно вызвать метод формы Close.

Наша программа должна выполнять два вида действий – загрузку файла для просмотра в многострочный текстовый редактор и запуск программы на выполнение. Для этого создадим две соответствующие процедуры OpenFile и ExecuteFile с одним входным параметром строкового типа для передачи имени файла. Процедура OpenFile должна открыть текстовый файл, считать все его строки и поместить их в многострочный редактор типа TMemo. Наиболее важные свойства компоненты данного типа приведены в табл. 9.

Процедура ExecuteFile должна запускать выполняемый файл. Для этого можно воспользоваться функцией WinExec, определенной в модуле Windows. Функция имеет два параметра. Первый типа PChar задает имя файла, второй целочисленный – способ запуска. В большинстве случаев способ запуска можно задавать равным 0. Тип данных PChar эквивалентен типу *char, используемому в языке C. Для приведения к нему обычной строки Delphi, например FileName типа string, можно просто записать PChar(FileName).



Рис. 18. Состав подменю, соответствующего первой лабораторной работе

Таблица 9. Основные свойства объектов типа TMemo

Свойство	Тип	Комментарий
Align	TAlign	Определяет режим автоматического изменения размера компоненты. Значение, равное alClient, позволяет автоматически растягивать компонент по размеру содержащего его визуального компонента, например формы.
ReadOnly	Boolean	При установке значения True, текст нельзя будет редактировать.
Lines	TStrings	Содержимое многострочного редактора.
ScrollBars	TScrollStyle	Определяет видимость полос прокрутки. Значение равное ssBoth включает обе полосы.

Теперь рассмотрим обработчики событий выбора пунктов меню «Открыть...» и «Выполнить...». В них нужно вызвать диалоги выбора файлов типа TOpenDialog и вызвать соответствующую процедуру OpenFile или ExecuteFile. Наиболее важные свойства компонента типа TOpenDialog приведены в табл. 10. Для того чтобы выдать диалог на экран, необходимо использовать его метод Execute, возвращающий логическое значение, которое равно True, если пользователь выбрал файл.

Теперь рассмотрим обработчики событий для пунктов меню, соответствующих файлам с расширениями dpr, pas и exe. Для первых двух логично написать один общий обработчик, который загружает текстовый файл. Предполагая, что все лабораторные работы находятся в своих каталогах, например, так, как на рис. 1, будем формировать имя файла автоматически, составляя его из некоторого каталога, содержащего все лабораторные работы, например, 'H:\Work\Delphi', затем

Замечание

Для назначения пунктам меню уже созданного обработчика событий необходимо в инспекторе объектов просто выбрать из ниспадающего списка нужный метод-обработчик.

Замечание

Для задания фильтра в компонентах для выбора файлов, можно вызвать редактор свойства Filter, нажав кнопку с многоточием справа от его значения в инспекторе объектов. В нем имеется таблица с двумя колонками: в левой помещается описание варианта фильтра, а в правой – сам фильтр.

имени главного меню, например, 'Lab1', и имени файла, указанного в тексте пункта меню, например, 'MainUnit.pas'. Основные процедуры в обоих обработчиках те же процедуры OpenFile и ExecuteFile.

В листингах 11-12 приведены тексты файлов проекта и модуля.

Таблица 10. Основные свойства объектов типа TOpenDialog

Свойство	Тип	Комментарий
DefaultExt	String	Задаёт расширение файла по умолчанию. Оно будет добавлено к введённому пользователем имени файла, если оно указано без расширения.
FileName	String	Имя выбранного файла.
Filter	String	В этом свойстве указывается список типов видимых в диалоге файлов. Например, в данной работе для команды «Открыть...» уместно указать в строке инспектора объектов (не входя в редактор свойства) значение 'Проекты Delphi (*.dpr) *.dpr Модули Delphi (*.pas) *.pas Текстовые файлы (*.txt) *.txt Все файлы *.*', а для команды «Выполнить...» – 'Выполняемые файлы *.exe;*.com;*.bat;*.pif'.
FilterIndex	Integer	Номер фильтра при открытии диалога
Options	TOpenOptions	В этом свойстве указываются различные параметры диалога. В нашем случае необходимо указать опцию ofFileMustExist для того чтобы пользователь не мог ввести имя несуществующего файла.
Title	String	Текст заголовка диалога выбора файлов.

Листинг 11. Текст файла проекта Lab5.dpr

```
program Lab5;

uses
  Forms,
  MainUnit in 'MainUnit.pas' {MainForm};

{$R *.RES}

begin
  Application.Initialize;
  Application.CreateForm(TMainForm, MainForm);
  Application.Run;
end.
```

Листинг 12. Текст главного модуля MainUnit.pas

```
unit MainUnit;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  Menus, StdCtrls;

const { Здесь указывается каталог с лабораторными работами }
  Root = 'H:\Work\Delphi\';

type
  TMainForm = class(TForm)
    MainMenu: TMainMenu;
    Lab11: TMenuItem;
    Lab21: TMenuItem;
    Lab31: TMenuItem;
    Lab41: TMenuItem;
    Lab51: TMenuItem;
    ItemExit: TMenuItem;
    Lab1dpr1: TMenuItem;
    N2: TMenuItem;
    MainUnitpas1: TMenuItem;
    N3: TMenuItem;
    Lab1exe1: TMenuItem;
    Lab1dpr2: TMenuItem;
    N4: TMenuItem;
    MainUnitpas2: TMenuItem;
    N5: TMenuItem;
    Lab1exe2: TMenuItem;
    Lab1dpr3: TMenuItem;
    N6: TMenuItem;
    MainUnitpas3: TMenuItem;
    N7: TMenuItem;
    Lab1exe3: TMenuItem;
    Lab1dpr4: TMenuItem;
    N8: TMenuItem;
    MainUnitpas4: TMenuItem;
    N9: TMenuItem;
    Lab1exe4: TMenuItem;
    Lab1dpr5: TMenuItem;
    N10: TMenuItem;
    MainUnitpas5: TMenuItem;
    N11: TMenuItem;
```



```
Lablexe5: TMenuItem;
Inputpas1: TMenuItem;
ListInputpas1: TMenuItem;
GridInputpas1: TMenuItem;
Memo: TMemo;
N12: TMenuItem;
ItemOpen: TMenuItem;
N14: TMenuItem;
ItemExecute: TMenuItem;
OpenFileDialog: TOpenDialog;
ExecuteFileDialog: TOpenDialog;
procedure ItemExitClick(Sender: TObject);
procedure ItemOpenClick(Sender: TObject);
procedure ItemExecuteClick(Sender: TObject);
procedure OpenFileClick(Sender: TObject);
procedure ExecuteFileClick(Sender: TObject);
private
  procedure OpenFile(FileName: string); // Загрузить содержимое текстового файла
  procedure ExecuteFile(FileName: string); // Запустить выполняемый файл
end;

var
  MainForm: TMainForm;

implementation

{$R *.DFM}

procedure TMainForm.ItemExitClick(Sender: TObject);
begin // Выход из программы
  Close;
end;

procedure TMainForm.ItemOpenClick(Sender: TObject);
begin // Выбрана команда меню "Открыть..."
  if OpenFileDialog.Execute then
    OpenFile(OpenFileDialog.FileName);
end;

procedure TMainForm.ItemExecuteClick(Sender: TObject);
begin // Выбрана команда меню "Выполнить..."
  if ExecuteFileDialog.Execute then
    ExecuteFile(ExecuteFileDialog.FileName);
end;

procedure TMainForm.OpenFileClick(Sender: TObject);
begin // Обработчик команды открытия текстовых файлов с расширениями .dpr и .pas
  with Sender as TMenuItem do
    OpenFile(Root+Parent.Caption+'\'+Caption);
end;

procedure TMainForm.ExecuteFileClick(Sender: TObject);
begin // Обработчик команды запуска выполняемых файлов с расширениями .exe
  with Sender as TMenuItem do
    ExecuteFile(Root+Parent.Caption+'\'+Caption);
end;

procedure TMainForm.OpenFile(FileName: string);
var f: TextFile; s: string;
begin // Процедура загрузки текстовых файлов в компоненту TMemo
  AssignFile(f, FileName);
```

```
try
  Reset(f); // Открыть файл для чтения
try
  Memo.Lines.Clear;
  while not EOF(f) do
    begin // Пока не достигнут конец файла, читать построчно
      ReadLn(f,s);
      Memo.Lines.Add(s);
    end;
  finally
    CloseFile(f); // Закрыть файл
  end;
except
  ShowMessage('Ошибка загрузки файла '+FileName);
end;
end;

procedure TMainForm.ExecuteFile(FileName: string);
begin // Запуск выполняемого файла
  if WinExec(PChar(FileName), 0)<32 then
    ShowMessage('Ошибка запуска файла '+FileName);
end;

end.
```

Вопросы и задания для самостоятельной работы

1. Добавить возможность редактирования файлов в TМemo и последующего сохранения в выбираемый с помощью компоненты TSaveDialog файл.
2. Как назначить нескольким пунктам меню один и тот же обработчик событий?
3. Создайте пункт меню, выдающий в отдельном модальном окне информацию о программе (автор, дата создания и т.д.).
4. Создайте пункт меню, по которому на экран выдается в отдельном модальном окне статистика о текущем загруженном файле – количество строк и символов.
5. Добавьте новые пункты меню для запуска каких-нибудь наиболее распространенных программ Windows.

Учебное издание

Алексей Владимирович Скворцов
Поддубная Тамара Николаевна

Меню и диалоги

(методические указания к
лабораторной работе № 5)