



# Самоучитель

Анатолий Хомоненко  
Владимир Гофман

# Delphi

2-е издание



Визуальное конструирование

Построение форм, меню

Использование графики и мультимедиа

Технологии BDE, ADO, dbExpress

Создание приложений баз данных

Работа с электронной почтой  
и Web-документами



+ CD

**Анатолий Хомоненко  
Владимир Гофман**

**Самоучитель**  
**Delphi**  
**2-е издание**

Санкт-Петербург  
«БХВ-Петербург»  
2008

УДК 681.3.068+800.92Delphi  
ББК 32.973.26-018.1  
X76

**Хомоненко, А. Д.**

X76 Самоучитель Delphi. — 2-е изд., перераб. и доп. / А. Д. Хомоненко, В. Э. Гофман. — СПб.: БХВ-Петербург, 2008. — 576 с.: ил. + CD-ROM  
ISBN 978-5-9775-0175-0

Описываются интерфейс системы Delphi на примере версии 7.0, состав и характеристика элементов проекта приложения, приемы программирования на языке Delphi. Рассматриваются визуальные компоненты, используемые для создания интерфейса приложений; компоненты и техника работы с текстовой информацией, кнопками и переключателями, а также формами. Описываются развитые элементы интерфейса, использование графики и мультимедиа, работа с файлами и каталогами. Даются понятия, связанные с реляционными базами данных. Описывается создание приложений баз данных, подготовка отчетов с помощью генератора Rave Reports. Даются основы разработки приложений для работы в Интернете, работа с электронной почтой и Web-документами. На компакт-диске содержатся многочисленные листинги программ, приведенных в книге.

*Для начинающих программистов*

УДК 681.3.068+800.92Delphi  
ББК 32.973.26-018.1

### **Группа подготовки издания:**

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Игорь Шишигин</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Анна Кузьмина</i>
Компьютерная верстка	<i>Ольги Сергиенко</i>
Корректор	<i>Зинаида Дмитриева</i>
Дизайн серии	<i>Инны Тачиной</i>
Оформление обложки	<i>Елены Беляевой</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 20.03.08.  
Формат 70×100<sup>1/16</sup>. Печать офсетная. Усл. печ. л. 46,44.  
Тираж 3000 экз. Заказ №  
"БХВ-Петербург", 194354, Санкт-Петербург, ул. Есенина, 5Б.

Отпечатано с готовых диапозитивов  
в ГУП "Типография "Наука"  
199034, Санкт-Петербург, 9 линия, 12

ISBN 978-5-9775-0175-0

© Хомоненко А. Д., Гофман В. Э., 2008  
© Оформление, издательство "БХВ-Петербург", 2008

# Оглавление

Предисловие .....	11
<b>ЧАСТЬ I. ОСНОВНЫЕ СРЕДСТВА DELPHI .....</b>	<b>15</b>
<b>Глава 1. Среда Delphi.....</b>	<b>16</b>
Характеристика проекта .....	22
Состав проекта .....	22
Файл проекта .....	23
Файлы формы .....	24
Файлы модулей.....	27
Файл ресурсов .....	27
Параметры проекта .....	28
Компиляция и выполнение проекта .....	30
Разработка приложения .....	31
Простейшее приложение .....	32
Создание пользовательского интерфейса приложения.....	33
Определение функциональности приложения.....	38
Средства интегрированной среды разработки.....	42
Управление параметрами среды .....	42
Менеджер проектов .....	43
Встроенный отладчик .....	44
Обозреватель проекта .....	45
Хранилище объектов .....	46
Справочная система .....	48
<b>Глава 2. Язык программирования Delphi.....</b>	<b>51</b>
Основные понятия.....	51
Алфавит.....	51
Словарь языка.....	52
Структура программы.....	53
Комментарии .....	56
Типы данных.....	57
Операторы.....	58
Директивы компилятора.....	58
Простые типы данных .....	59
Целочисленные типы .....	59
Литерные типы .....	59
Логические типы .....	60

Интервальные типы.....	60
Вещественные типы.....	61
Структурные типы данных.....	61
Строки.....	61
Массивы.....	62
Множества.....	62
Записи.....	63
Файлы.....	65
Другие типы данных.....	66
Процедурные типы.....	66
Вариантные типы.....	66
Выражения.....	68
Арифметические выражения.....	69
Логические выражения.....	70
Строковые выражения.....	72
Простые операторы.....	74
Оператор присваивания.....	74
Оператор перехода.....	75
Пустой оператор.....	75
Оператор вызова процедуры.....	75
Структурированные операторы.....	76
Составной оператор.....	76
Условный оператор.....	76
Оператор выбора.....	77
Операторы цикла.....	78
Оператор доступа.....	81
Подпрограммы.....	82
Процедуры.....	84
Функции.....	85
Параметры и аргументы.....	86
Особенности объектно-ориентированного программирования.....	88
Основные концепции ООП.....	88
Классы и объекты.....	89
Поля.....	91
Свойства.....	92
Методы.....	92
Сообщения и события.....	93
Динамическая информация о типе.....	95
Библиотека визуальных компонентов.....	97
<b>Глава 3. Визуальные компоненты.....</b>	<b>100</b>
Страницы с визуальными компонентами.....	100
Базовый класс <i>TControl</i> .....	103
Свойства.....	104

События .....	113
Методы.....	117
<b>Глава 4. Работа с текстом .....</b>	<b>119</b>
Класс <i>TStrings</i> .....	119
Использование надписей .....	123
Однострочный редактор.....	125
Многострочный редактор.....	127
Общие элементы компонентов редактирования .....	129
Работа со списками .....	132
Простой список .....	133
Комбинированный список.....	135
Общие свойства и события списков .....	136
<b>Глава 5. Кнопки и переключатели .....</b>	<b>139</b>
Работа с кнопками.....	139
Стандартная кнопка .....	139
Кнопка с рисунком .....	142
Использование переключателей и флажков .....	145
Флажок .....	145
Переключатель .....	147
Объединение элементов управления .....	150
Группа .....	150
Панель .....	150
Область прокрутки .....	151
<b>Глава 6. Формы и диалоги .....</b>	<b>154</b>
Характеристики формы .....	154
Управление формами.....	157
Пример приложения с формой .....	166
Организация взаимодействия форм.....	170
Особенности модальных форм .....	172
Процедуры и функции, реализующие диалоги.....	175
Стандартные диалоги.....	177
Шаблоны форм.....	182
<b>Глава 7. Панель инструментов, меню и механизм действий .....</b>	<b>184</b>
Характеристика меню .....	184
Главное меню .....	186
Контекстное меню.....	187
Конструктор меню .....	188
Динамическая настройка меню.....	189
Панели инструментов .....	190
Создание панели инструментов на основе компонента <i>Panel</i> .....	191
Создание панели инструментов на основе компонента <i>Form</i> .....	200

Характеристика механизма действий.....	200
Стандартные действия.....	203
Менеджер действий.....	204
<b>ЧАСТЬ II. РАЗВИТЫЕ СРЕДСТВА DELPHI.....</b>	<b>207</b>
<b>Глава 8. Управление приложением и экраном.....</b>	<b>208</b>
Объект <i>Application</i> .....	208
Объект <i>Screen</i> .....	219
<b>Глава 9. Обработка исключений.....</b>	<b>222</b>
Виды ошибок.....	222
Классы исключений.....	225
Обработка исключений.....	228
Глобальная обработка.....	228
Локальная обработка.....	230
<b>Глава 10. Развитые элементы интерфейса.....</b>	<b>237</b>
Диапазоны значений.....	237
Реверсивные счетчики.....	242
Компонент <i>UpDown</i> .....	242
Компонент <i>SpinButton</i> .....	245
Компонент <i>SpinEdit</i> .....	246
Строка состояния.....	246
Элементы с вкладками.....	249
Одностраничный блокнот.....	249
Многостраничный блокнот.....	254
<b>Глава 11. Организация приложений.....</b>	<b>255</b>
Создание многодокументных приложений.....	256
Особенности многодокументных приложений.....	256
Пример многодокументного приложения.....	262
Шаблон многодокументного приложения.....	269
Особенности консольного приложения.....	270
Запуск других приложений.....	271
<b>Глава 12. Работа с графикой.....</b>	<b>275</b>
Возможности рисования при выполнении программы.....	277
Поверхность рисования.....	278
Графические компоненты.....	289
Компонент <i>Shape</i> .....	290
Компонент <i>Bevel</i> .....	291
Компонент <i>Image</i> .....	292
Компонент <i>PaintBox</i> .....	295
Компонент <i>ImageList</i> .....	295

<b>Глава 13. Работа с мультимедиа .....</b>	<b>300</b>
Воспроизведение видеоклипов .....	301
Управление мультимедийными устройствами .....	304
<b>Глава 14. Работа с файлами и каталогами.....</b>	<b>315</b>
Средства системных модулей .....	315
Компоненты для работы с файлами и каталогами .....	325
Компонент <i>DriveComboBox</i> .....	326
Компонент <i>DirectoryListBox</i> .....	327
Компонент <i>FileListBox</i> .....	328
Компонент <i>FilterComboBox</i> .....	330
Пример приложения.....	331
<b>ЧАСТЬ III. РАБОТА С БАЗАМИ ДАННЫХ.....</b>	<b>337</b>
<b>Глава 15. Введение в базы данных .....</b>	<b>338</b>
Основные понятия.....	338
Банки данных.....	338
Модели данных.....	339
Базы данных и приложения .....	340
Варианты архитектуры .....	341
Реляционные базы данных .....	342
Таблицы баз данных .....	342
Ключи и индексы .....	345
Способы доступа к данным .....	348
Связь между таблицами.....	348
Механизм транзакций.....	352
Бизнес-правила .....	353
Форматы таблиц .....	353
Инструменты для работы с базами данных .....	356
Исключения баз данных .....	358
Этапы создания информационной системы.....	360
Создание таблиц базы данных .....	361
Описание полей .....	363
Задание индексов .....	364
Задание ограничений на значения полей .....	367
Задание ссылочной целостности.....	367
Задание паролей .....	367
Задание языкового драйвера .....	368
Изменение структуры таблицы .....	368
<b>Глава 16. Компоненты для работы с базами данных .....</b>	<b>369</b>
Характеристика приложения для работы с базами данных.....	369
Источник данных .....	371

Визуальные компоненты .....	372
Отображение и редактирование значения строкового поля .....	374
Отображение и редактирование значения логического поля .....	375
Представление записей с помощью сетки .....	376
Столбцы сетки .....	378
Использование навигатора .....	384
Наборы данных .....	386
Общая характеристика .....	386
Способы доступа к данным .....	389
Состояния и режимы .....	389
Объекты-поля .....	394
Редактор полей .....	395
Операции с полями .....	398
<b>Глава 17. Технология BDE .....</b>	<b>402</b>
Общая характеристика .....	402
Особенности компонента <i>Table</i> .....	403
Особенности компонента <i>Query</i> .....	406
Операции с данными .....	410
Сортировка записей .....	411
Навигация по набору данных .....	413
Фильтрация записей .....	416
Поиск записей .....	420
Модификация набора данных .....	423
Пример формы приложения .....	433
Связывание таблиц .....	440
<b>Глава 18. Технология ADO .....</b>	<b>444</b>
Общая характеристика .....	444
Установление соединения .....	446
Управление соединением и транзакциями .....	450
Компоненты доступа к данным .....	452
Доступ к таблицам .....	454
Выполнение запросов .....	455
Вызов хранимых процедур .....	455
Компонент <i>ADODataset</i> .....	456
Команды ADO .....	457
Пример приложения .....	458
<b>Глава 19. Технология dbExpress .....</b>	<b>462</b>
Общая характеристика .....	462
Установление соединения с сервером .....	464
Компоненты доступа к данным .....	467
Универсальный доступ к данным .....	468
Просмотр таблиц .....	473

Выполнение SQL-запроса.....	474
Выполнение хранимых процедур .....	475
Компонент редактирования набора данных .....	475
Отладка соединения с сервером .....	479

## **Глава 20. Подготовка отчетов ..... 481**

Характеристика генератора отчетов.....	481
Визуальное конструирование отчетов.....	483
Интерфейс визуального конструктора .....	483
Состав проекта отчетов .....	484
Редактор событий.....	487
Компоненты, представленные на многостраничной панели инструментов.....	488
Компоненты отображения данных .....	491
Компоненты управления отчетом.....	493
Компонент-проект отчета.....	494
Компонент управления отчетом .....	495
Компоненты установления соединения.....	495
Схема управления отчетом и подсоединения данных.....	496
Примеры создания и просмотра отчетов .....	498
Предварительный просмотр отчета .....	499
Простой отчет приложения базы данных.....	501

## **ЧАСТЬ IV. DELPHI И ИНТЕРНЕТ ..... 505**

### **Глава 21. Основные элементы интернет-технологий..... 506**

Сценарии JavaScript, JScript и VBScript .....	507
Элементы управления ActiveX.....	508
Апплеты и сервлеты Java.....	509
Интерфейсы CGI и WinCGI .....	509
Интерфейсы ISAPI/NSAPI.....	510
Страницы ASP, PHP и IDC/HTX .....	511
Формирование Web-страниц.....	512
Интерфейсы OLE DB, ADO, ODBC .....	512
Статическая публикация БД.....	513
Динамическая публикация БД .....	514
Web-приложения .....	515
Протоколы передачи данных .....	516
Универсальный указатель ресурсов (URL).....	516
Язык разметки гипертекста HTML.....	517
Расширяемый язык разметки XML .....	518

### **Глава 22. Web-приложения и интерфейсы ..... 520**

Характеристика Web-приложений .....	520
Web-приложения в сетях интранет.....	521

Web-приложения с модулями расширения серверной части .....	523
Web-приложения с модулями расширения клиентской части .....	526
Архитектура Web-приложений, публикующих БД .....	527
Двухуровневые Web-приложения .....	527
Трехуровневые Web-приложения .....	528
Интерфейсы Web-приложений .....	530
Общий интерфейс взаимодействия CGI .....	530
Интерфейс программирования серверных приложений ISAPI .....	535
<b>Глава 23. Работа с электронной почтой .....</b>	<b>539</b>
Использование функции <i>ShellExecute</i> .....	539
Использование интерфейса MAPI .....	540
<b>Глава 24. Работа с Web-документами .....</b>	<b>546</b>
Характеристика компонента <i>WebBrowser</i> .....	546
Управление с помощью процедуры <i>ExecWB</i> .....	550
Работа в режиме HTML-редактора .....	550
Пример формы приложения .....	552
<b>Глава 25. Публикация баз данных .....</b>	<b>557</b>
Компоненты, используемые при разработке Web-приложений .....	557
Статическая публикация .....	559
Компоненты генерации HTML-страниц .....	564
<b>ПРИЛОЖЕНИЯ .....</b>	<b>565</b>
<b>Приложение 1. Взаимосвязь программ с Delphi .NET .....</b>	<b>566</b>
Характеристика платформы Microsoft .NET .....	566
Типы приложений Delphi .NET .....	567
Перенос приложений в Delphi .NET .....	569
<b>Приложение 2. Описание компакт-диска .....</b>	<b>570</b>
<b>Предметный указатель .....</b>	<b>571</b>

# Предисловие

В настоящее время среди широкого круга пользователей по-прежнему популярна система объектно-ориентированного программирования Delphi, основу которой составляет язык Delphi (в прошлом называемый Object Pascal). Delphi позволяет быстро создавать приложения различной степени сложности на основе применения технологии визуального программирования.

Книга посвящена основам работы с Delphi и освоению приемов программирования с использованием визуальных средств. В ней рассматриваются важнейшие средства Delphi, технология создания приложений для решения важнейших общих задач (от простейших программ до не очень сложных приложений для работы с базами данных и Интернетом), с которыми приходится сталкиваться на начальном этапе освоения системы программирования.

В книге рассматривается большое число примеров, демонстрирующих основные возможности Delphi. Примеры взяты из работающих программ, которые читатель может использовать в своих разработках. Нами рассматривается использование наиболее общих языковых средств, присутствующих в последних версиях системы, предшествующих Delphi .NET. Как следствие, книга не привязана к конкретной версии, и приведенные примеры, как правило, работоспособны для 4—7 версий Delphi. При описании интерфейса использовалась 7-я версия системы.

Книга состоит из четырех частей и двух приложений.

**Часть I. "Основные средства Delphi".** Содержит описание интерфейса системы программирования, состав и характеристику элементов проекта приложения. Описывается язык программирования Delphi: типы данных, основные конструкции языка, важнейшие приемы программирования, понятия объектно-ориентированного программирования.

Рассматриваются важнейшие визуальные компоненты, используемые для создания интерфейса приложений. При этом дается состав страниц Палитры компонентов, содержащих визуальные компоненты; описывается класс

TControl, который является базовым для большинства визуальных компонентов и включает в себя общие для визуальных компонентов свойства, события и методы.

Рассматриваются компоненты и техника работы с информацией (текстом) по ее отображению, вводу и редактированию. С этой целью описывается класс TStrings, являющийся базовым классом для операций со строковыми данными; компонент Label, служащий для отображения надписей (текста, используемого в качестве заголовков для некоторых управляющих элементов); компоненты Edit, MaskEdit, Memo и RichEdit, обеспечивающие возможности редактирования информации; средства и техника работы со списками.

Описываются компоненты и техника работы с кнопками и переключателями. Освещается техника объединения, или группирования различных элементов управления, которая может понадобиться, например, при работе с переключателями на форме или при создании панели инструментов.

Рассматриваются компоненты и техника создания форм, являющихся важнейшим визуальным компонентом, центральной частью практически любого приложения и представляющих собой видимые окна Windows. Приводятся характеристики формы, приемы организации взаимодействия форм. Описываются процедуры и функции, используемые для организации предопределенных (стандартных) диалогов, которые также являются формами и служат для выполнения типовых действий, таких как вывод сообщений и ввод информации. Рассматриваются технологии создания меню, панелей инструментов и настройки интерфейса приложения с помощью механизма действий.

**Часть II. "Развитые средства Delphi".** Рассматривается организация управления приложением и экраном. Описываются важные для отладки приложений Delphi вопросы обработки исключительных ситуаций, связанных с ошибками при выполнении программ. При этом характеризуются виды ошибок и классы исключений, рассматриваются варианты глобальной и локальной обработки исключений, вызов исключений, особенности отладки обработчиков исключений.

Рассматриваются вопросы создания развитых элементов интерфейса приложений: работа с диапазонами значений, организация счетчиков, варианты создания панелей инструментов с помощью широко используемых компонентов (Panel, ToolBar, CoolBar, Form), задание таблиц, а также одностраничных и многостраничных блокнотов. Описываются вопросы организации приложений: создание многодокументных приложений (особенности, пример построения и шаблон), особенности создания консольного приложения, техника запуска других приложений.

Освещаются вопросы работы с графикой: рисование в процессе выполнения программ, используемые важнейшие графические компоненты Shape, Bevel,

Image, техника и средства построения индикаторов и диаграмм. Описываются вопросы воспроизведения видеоклипов и управления мультимедийными устройствами. Рассматриваются компоненты для работы с файлами и каталогами и примеры их использования.

**Часть III. "Работа с базами данных".** Рассматриваются основные понятия баз данных; характеризуются элементы реляционных баз данных и техника их использования (таблицы, ключи и индексы, способы доступа к данным, связь между таблицами, механизм транзакций и др.); дается обзор средств и компонентов для работы с базами данных, а также механизмов построения приложений; характеризуется технология создания информационной системы; описывается создание таблиц и приложений баз данных для технологий BDE, ADO и dbExpress.

Описываются основные компоненты для работы с базами данных: источник данных (служащего промежуточным звеном между набором данных и визуальными компонентами), визуальные компоненты, с помощью которых пользователь управляет набором данных (для навигации по набору данных, а также для отображения и редактирования записей), а также наборы данных.

При этом рассматриваются общие свойства наборов данных, используемых для выполнения операций над данными таблиц. Освещаются состояния и режимы наборов данных, доступ к полям. Отмечаются особенности важнейших наборов данных Table и Query. Описываются невидимые объекты типа TField, служащие для доступа к данным полей записей набора данных, работа с Редактором полей и доступ к значениям полей.

Работа с данными рассматривается на примере использования навигационного способа доступа к локальным БД с помощью процессора баз данных BDE, допускающего использование наборов данных Table или Query. При этом освещаются важнейшие операции, реализуемые при навигационном способе доступа: сортировка записей в наборе данных, навигация по набору данных, редактирование записей, вставка и удаление записей, фильтрация записей. Здесь же описывается работа со связанными таблицами.

Рассматривается создание приложений с использованием технологии ADO: установление соединения; управление соединениями и транзакциями; особенности компонентов наборов данных, обеспечивающих доступ к таблицам, выполнение SQL-запросов, хранимых процедур и др. Приводится пример приложения, обеспечивающего доступ к различным таблицам из базы данных, их редактирование и сохранение внесенных изменений на сервере.

Описывается технология dbExpress, которая обеспечивает построение легких (по объему кода) приложений для работы с базами данных. Рассматривается установление соединения с сервером, дается характеристика основ-

ных компонентов доступа к данным (в том числе универсальный доступ к данным, работа с таблицами, запросами и др.), отладка соединения с сервером.

Рассматривается подготовка отчетов (печатных документов, содержащих данные, аналогичные получаемым в результате выполнения запроса к базе данных) с помощью генератора Rave Reports. При этом описываются компоненты, предназначенные для создания отчетов; процедура печати отчета; технология подготовки простого отчета. Приводятся примеры создания отчетов.

**Часть IV. "Delphi и Интернет".** Рассматриваются основные понятия интернет-технологий, на которые опирается разработчик приложений, взаимодействующих с сетью Интернет. Дается обзор вариантов построения Web-приложений. Описываются важнейшие интерфейсы, используемые при разработке Web-приложений: общий интерфейс взаимодействия CGI и интерфейс программирования серверных приложений ISAPI.

Приводятся примеры создания приложений, взаимодействующих с Интернетом: для работы с электронной почтой, для работы с Web-документами с помощью обозревателя Интернет — компонента WebBrowser и для публикации баз данных.

В *приложении 1* дается краткое описание системы программирования Delphi .NET, указываются типы создаваемых с ее помощью приложений, а также затрагивается вопрос переноса обычных приложений Delphi в среду Delphi .NET. В *приложении 2* приводится описание компакт-диска, прикладываемого к данной книге.

Авторы выражают признательность Мещерякову Е. В. за помощь в подготовке материалов, посвященных работе в сети Интернет, а также Шишигину И. В. за конструктивные советы по формированию состава книги и помощь в поиске наглядных примеров по работе в системе программирования Delphi.

Книга ориентирована на начинающих программистов.



# ЧАСТЬ I

## Основные средства Delphi

**Глава 1.** Среда Delphi

**Глава 2.** Язык программирования Delphi

**Глава 3.** Визуальные компоненты

**Глава 4.** Работа с текстом

**Глава 5.** Кнопки и переключатели

**Глава 6.** Формы и диалоги

**Глава 7.** Панель инструментов, меню и механизм действий

# ГЛАВА 1



## Среда Delphi

Прикладные программы, или приложения, Delphi создаются в *интегрированной среде разработки* (Integrated Development Environment, IDE). Пользовательский интерфейс этой среды служит для организации взаимодействия с программистом и включает в себя ряд окон, содержащих различные элементы управления. С помощью средств интегрированной среды разработчику удобно проектировать интерфейсную часть приложения, а также писать программный код и связывать его с элементами управления. В интегрированной среде разработки приложения проходят все этапы создания, включая отладку.

Интегрированная среда разработки Delphi представляет собой многооконную систему. Вид интегрированной среды разработки (пользовательский интерфейс) может различаться в зависимости от настроек. После загрузки интерфейс Delphi выглядит так, как показано на рис. 1.1, и первоначально включает шесть окон:

- главное окно (**Delphi - Project1**);
- окно Обзорщика дерева объектов (**Object TreeView**);
- окно Инспектора объектов (**Object Inspector**);
- окно Формы, или Конструктора формы (**Form1**);
- окно Редактора кода (**Unit1.pas**);
- окно Проводника кода (**Exploring Unit1.pas**).

Последние два окна находятся позади окна Формы, причем окно Проводника кода пристыковано слева к окну Редактора кода, поэтому оба этих окна имеют общий заголовок **Unit1.pas**.

На экране кроме указанных окон могут присутствовать и другие окна, отображаемые при вызове соответствующих средств, например, окно Редактора изображений (**Image Editor**). Окна Delphi можно перемещать, изменять их

размеры и убирать с экрана (кроме главного окна), а также состыковывать между собой.

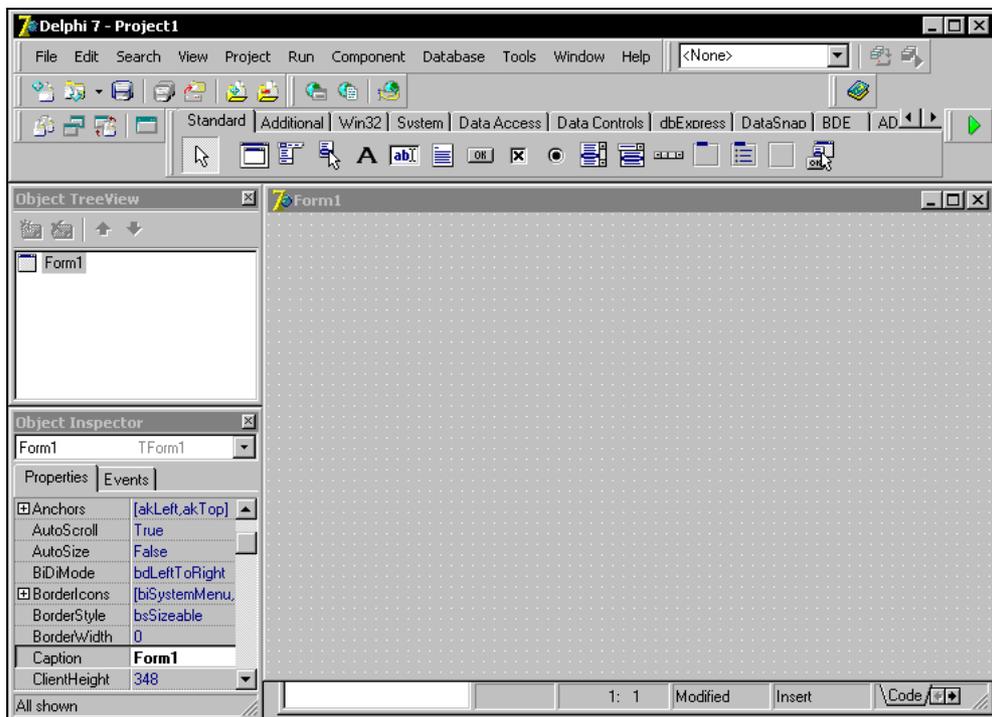


Рис. 1.1. Вид интегрированной среды разработки

Несмотря на наличие многих окон, Delphi является однодокументной средой и позволяет одновременно работать только с одним приложением (проектом приложения). Название проекта приложения выводится в строке заголовка главного окна в верхней части экрана.

При сворачивании главного окна сворачивается весь интерфейс Delphi и, соответственно, все открытые окна; при закрытии главного окна работа с Delphi прекращается. Главное окно Delphi включает:

- главное меню;
- панели инструментов;
- Палитру компонентов.

Главное меню содержит обширный набор команд для доступа к функциям Delphi, основные из которых рассматриваются при изучении связанных с этими командами операций.

Панели инструментов находятся под главным меню в левой части главного окна и содержат пятнадцать кнопок для вызова наиболее часто используемых команд главного меню, например, **File | Open** (Файл | Открыть) или **Run | Run** (Выполнение | Выполнить).

Вызвать многие команды главного меню можно также с помощью комбинаций клавиш, указываемых справа от названия соответствующей команды. Например, команду **Run | Run** можно вызвать с помощью клавиши <F9>, а команду **View | Units** (Просмотр | Модули) — с помощью комбинации клавиш <Ctrl>+<F12>.

Всего имеется 6 панелей инструментов:

- Standard** (Стандартная);
- View** (Просмотр);
- Debug** (Отладка);
- Custom** (Пользователь);
- Desktop** (Рабочий стол);
- Internet** (Интернет).

Отображением панелей инструментов и настройкой кнопок на них можно управлять. Эти действия выполняются с помощью контекстного меню панелей инструментов, вызываемого щелчком правой кнопки мыши при размещении указателя в области панелей инструментов или главного меню.

С помощью контекстного меню можно также управлять видимостью Палитры компонентов (**Component Palette**).

*Палитра компонентов* находится под главным меню в правой части главного окна и содержит множество компонентов, размещаемых в создаваемых формах. *Компоненты* являются своего рода строительными блоками, из которых конструируются формы приложения. Все компоненты разбиты на группы, каждая из которых в Палитре компонентов располагается на отдельной странице, а сами компоненты представлены значками. Нужная страница Палитры компонентов выбирается щелчком мыши на ее значке. К числу основных страниц Палитры компонентов можно отнести следующие:

- Standard** — стандартная;
- Additional** — дополнительная;
- Win32** — 32-разрядного интерфейса Windows;
- System** — доступа к системным функциям;
- Data Access** — работы с информацией из баз данных;
- Data Controls** — создания элементов управления данными;

- ❑ **dbExpress** — доступа к SQL-серверам;
- ❑ **DataSnap** — создания многоуровневых приложений баз данных;
- ❑ **BDE** — доступа к данным с помощью BDE;
- ❑ **ADO** — связи с базами данных с использованием объектов данных ActiveX;
- ❑ **Interbase** — обеспечения непосредственного доступа к одноименной базе данных;
- ❑ **WebServices** — создания клиентских приложений, использующих Web-сервис с помощью технологии SOAP;
- ❑ **InternetExpress** — создания приложений Internet Express, которые являются одновременно Web-сервером и клиентом распределенной базы данных;
- ❑ **Internet** — создания приложений для работы в Интернете;
- ❑ **Dialogs** — создания стандартных диалоговых окон;
- ❑ **ActiveX** — компонентов ActiveX;
- ❑ **RAVE** — генерации отчетов в приложении;
- ❑ **Indy Clients** — платформо-независимые компоненты Интернета для клиента;
- ❑ **IW Standard** — стандартная для работы в Интернете.

Окно Формы (или Конструктора формы) первоначально находится в центре экрана и имеет заголовок **Form1**. В нем выполняется проектирование формы, в процессе которого в форму из Палитры компонентов помещаются необходимые компоненты. При этом проектирование заключается в визуальном конструировании формы, а действия разработчика похожи на работу в среде простого графического редактора. Сам Конструктор формы во время ее проектирования остается как бы "за кадром", и разработчик имеет дело непосредственно с формой, поэтому часто окно Конструктора формы также называют окном Формы или просто "формой".

Окно Редактора кода (**Unit1.pas**) после запуска системы программирования находится под окном Формы и почти полностью перекрывается им. Редактор кода представляет собой обычный текстовый редактор, с помощью которого можно редактировать текст модуля и другие текстовые файлы приложения (например, файл проекта). Каждый редактируемый файл находится в окне Редактора кода на отдельной странице, доступ к которой осуществляется щелчком на соответствующем значке. Первоначально в окне Редактора кода на странице **Code** содержится одна вкладка **Unit1** исходного кода модуля формы `Form1` разрабатываемого приложения.

Переключаться между окнами Формы и Редактора кода удобно с помощью клавиши <F12>.

Окно Проводника кода (**Exploring Unit1.pas**) пристыковано слева к окну Редактора кода. В нем в виде дерева отображаются все объекты модуля формы, например переменные и процедуры (рис. 1.2). В окне Проводника кода удобно просматривать объекты приложения и быстро переходить к нужным объектам, что особенно важно для больших модулей. Окно Проводника кода открывается командой **Code Explorer** (Проводник кода) меню **View** (Просмотр).

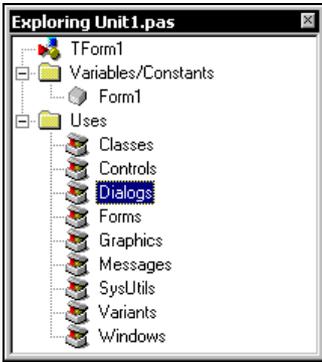


Рис. 1.2. Окно Проводника кода

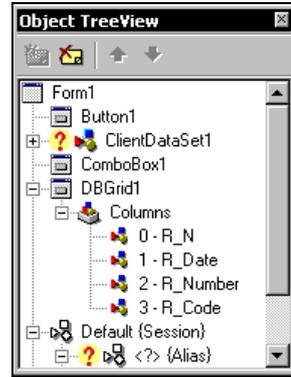


Рис. 1.3. Окно Обзорера дерева объектов

Окно Обзорера дерева (**Object TreeView**) объектов после запуска системы находится под главным окном и отображает древовидную структуру объектов текущей формы (первоначально `Form1`). Это окно удобно использовать в случае форм, служащих для обработки баз данных, т. к. оно позволяет изменять связи между компонентами, например, переназначить таблице источник данных другой таблицы. На рис. 1.3 показана структура объектов формы `Form1`, предназначенной для работы с таблицей базы данных.

Окно Инспектора объектов (**Object Inspector**) находится под окном Обзорера дерева объектов в левой части экрана и отображает свойства и события объектов для текущей формы `Form1`. Его можно открыть командой **View | Object Inspector** (Просмотр | Инспектор объектов) или нажатием клавиши <F11>.

Окно Инспектора объектов имеет две страницы: **Properties** (Свойства) и **Events** (События).

Страница **Properties** отображает информацию о текущем (выбранном) компоненте в окне Формы и при проектировании формы позволяет удобно изменять многие свойства компонентов.

Страница **Events** определяет процедуру, которую компонент должен выполнить при возникновении указанного события. Если для какого-либо события задана такая процедура, то в процессе выполнения приложения при возникновении этого события процедура вызывается автоматически. Такие процедуры служат для обработки соответствующих событий, поэтому их называют *процедурами* — *обработчиками событий* или просто *обработчиками*. Отметим, что события также являются свойствами, которые указывают на свои обработчики.

В конкретный момент времени Инспектор объектов показывает свойства и события текущего (выбранного) компонента, имя и тип которого отображаются в списке под заголовком окна Инспектора объектов. Компонент, расположенный в форме, можно выделить щелчком мыши на нем или выбором в списке Инспектора объектов. У каждого компонента есть набор свойств и событий, определяющих его особенности.

Инспектор объектов позволяет группировать свойства и события по категориям или в алфавитном порядке. Свойства (и их значения) отображаются различными цветами. В Инспекторе объектов содержатся и свойства, предназначенные только для чтения. Кроме того, можно настраивать свойства Инспектора объектов.

По умолчанию Инспектор объектов отображает названия свойств и событий в алфавитном порядке (см. рис. 1.1). Отображение их по категориям выполняется командой **Arrange | by Category** (Расположить | По категориям) контекстного меню Инспектора объектов.

По умолчанию Инспектор объектов отображает все свойства и события объектов. Можно отключить/включить отображение некоторой категории, убрав/установив отметку в соответствующем пункте (например, **Action**) подменю команды **View** контекстного меню.

Delphi поддерживает технологию *dock-окон*, которые могут стыковаться (соединяться) друг с другом с помощью мыши. Такими окнами являются инструментальные (недиалоговые) окна интегрированной среды разработки, в том числе окна Инспектора объектов и Проводника кода. Состыкованные окна удобно, например, перемещать по экрану или изменять их размеры.

Для соединения двух окон следует с помощью мыши поместить одно из них на другое и после изменения вида рамки перемещаемого окна отпустить его, после чего это окно автоматически пристыкуется сбоку ко второму окну. Разделение окон выполняется перемещением пристыкованного окна за двойную линию, размещенную под общим заголовком. После соединения окна представляют собой одно общее окно, разделенное на несколько частей. При стыковке/отстыковке окно изменяет свое название. Так, окно Проводника кода, состыкованное с окном Редактора кода, имеет общее с ним название,

например, **Unit1.pas**, в то время как при отстыковке название изменяется на **Exploring Unit1.pas**. Окна Инспектора объектов и Обзорщика дерева объектов при стыковке объединяют свои названия (названия всех окон указываются через запятую).

Для окон Инспектора объектов и Обзорщика дерева объектов можно установить режим **Stay on Top** (Расположить наверху), расположив их поверх других окон. Это выполняется включением одноименной отметки в контекстном меню.

Скрытое окно вызывается на экран командой меню **View**. Например, окно Проводника кода выводится на экран командой **View | Code Explorer**.

## Характеристика проекта

В этом разделе рассматриваются: состав проекта, файл проекта, файлы форм, файлы модулей, файл ресурсов и параметры проекта.

### Состав проекта

Создаваемое в среде Delphi приложение состоит из нескольких элементов, объединенных в *проект*. В состав проекта входят следующие элементы (в скобках указаны расширения имен файлов):

- код проекта (dpr);
- описания форм (dfm — для Windows, xfm — кроссплатформенный вариант);
- модули и модули форм (pas);
- параметры проекта (dof — для Windows, kof — для Linux);
- параметры среды (cfg);
- описание ресурсов (res).

Взаимосвязи между отдельными частями (файлами) проекта показаны на рис. 1.4.

Кроме приведенных файлов, автоматически могут создаваться и другие файлы, например, резервные копии файлов: ~dpr — для файлов с расширением dpr; ~pa — для файлов с расширением pas.

При запуске Delphi автоматически создается новый проект с именем Project1, отображаемым в заголовке главного окна Delphi. Этот проект имеет в своем составе одну форму Form1, название которой видно в окне Формы. Разработчик может изменить предлагаемое по умолчанию имя проекта, а также установить параметры среды таким образом, что после загрузки Delphi

будет автоматически загружаться приложение, разработка которого выполнялась в последний раз.

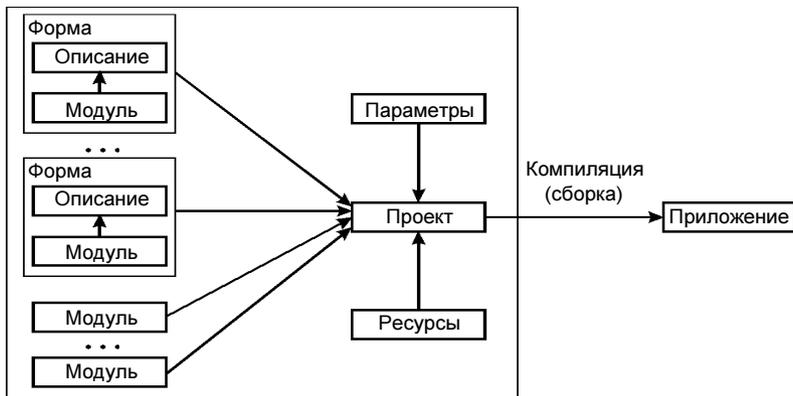


Рис. 1.4. Связи между файлами проекта

Обычно файлы проекта располагаются в одном каталоге. Поскольку даже относительно простой проект включает достаточно много файлов, а при добавлении к проекту новых форм количество этих файлов увеличивается, для каждого нового проекта целесообразно создавать отдельный каталог, где и сохранять все файлы проекта.

## Файл проекта

Файл проекта является центральным файлом проекта и представляет собой собственно программу. Для приложения, имеющего в составе одну форму, файл проекта имеет следующий вид:

```

program Project1;
uses
  Forms,
  Unit1 in 'Unit1.pas' {Form1};
{$R *.res}
begin
  Application.Initialize;
  Application.CreateForm(TForm1, Form1);
  Application.Run;
end.
  
```

Имя проекта (программы) совпадает с именем файла проекта и указывается при сохранении этого файла на диске, первоначально это имя `Project1`. То же имя имеют файлы ресурсов и параметров проекта, при переименовании файла проекта данные файлы переименовываются автоматически.

Сборка всего проекта выполняется при компиляции файла проекта. При этом имя создаваемого приложения (exe-файл) или динамически загружаемой библиотеки (dll-файл) совпадает с названием файла проекта.

В дальнейшем мы будем подразумевать, что создается приложение, а не динамически загружаемая библиотека.

В разделе `uses` указывается имя подключаемого модуля `Forms`, который является обязательным для всех приложений, имеющих в своем составе формы. Кроме того, в разделе `uses` перечисляются подключаемые модули всех форм проекта, первоначально это модуль `Unit1` формы `Form1`.

Директива `$R` подключает к проекту файл ресурсов, имя которого по умолчанию совпадает с именем файла проекта. Поэтому вместо имени файла ресурса указан символ `*`. Кроме этого файла разработчик может подключить к проекту и другие ресурсы, самостоятельно добавив директивы `$R` и указав в них соответствующие имена файлов ресурсов.

Программа проекта содержит всего три оператора, выполняющих инициализацию приложения, создание формы `Form1` и запуск приложения. Эти операторы будут рассмотрены в следующих главах.

При выполнении разработчиком каких-либо операций с проектом Delphi формирует код файла проекта автоматически. Например, при добавлении новой формы в файл проекта добавляются две строки кода, относящиеся к этой форме, а при исключении формы из проекта эти строки автоматически исключаются. При необходимости программист может вносить изменения в файл проекта самостоятельно, однако подобные действия могут разрушить целостность проекта и поэтому обычно выполняются только опытными программистами. Отметим, что некоторые операции, например, создание обработчика события для объекта `Application`, системой Delphi автоматически не выполняются и требуют самостоятельного кодирования в файле проекта.

Отображение кода файла проекта в окне Редактора кода задается командой **Project | View Source** (Проект | Просмотр источника).

В файле проекта для многих приложений имеется похожий код, поэтому в дальнейшем при рассмотрении большинства приложений содержимое этого файла не приводится.

## Файлы формы

Для каждой формы в составе проекта автоматически создаются файл описания формы (расширение `dfm`) и файл модуля формы (расширение `pas`).

*Файл описания формы* является ресурсом Delphi и содержит характеристики формы и ее компонентов. Разработчик обычно управляет этим файлом через

окна Формы и Инспектора объектов. При конструировании формы в файл описания автоматически вносятся соответствующие изменения. Содержимое файла описания формы определяет ее вид. При необходимости можно отобразить этот файл на экране в текстовом виде, что выполняется командой **View as Text** (Просмотреть как текст) контекстного меню формы. При этом окно Формы пропадает с экрана, а содержимое файла описания формы открывается в окне Редактора кода и доступно для просмотра и редактирования.

Отметим, что в начальной (пустой) форме типа `TForm1` отсутствуют строки, относящиеся к кнопке `Button1` (выделены полужирным начертанием).

Файл описания содержит перечень всех объектов формы, включая саму форму, а также свойства этих объектов. Для каждого объекта указывается его тип; для формы ее тип (класс) `TForm1` описывается в модуле этой формы. Если в строчке `Caption = 'Form1'`, определяющей заголовок формы, вместо `Form1` ввести, например, текст *Первая форма*, то заголовок формы изменится на новый. Однако на практике подобные действия обычно выполняются в окне Инспектора объектов.

Повторное открытие окна формы выполняется командой **View | Forms** (Просмотр | Формы) или комбинацией клавиш `<Shift>+<F12>`, после чего открывается диалоговое окно **View Form** (Просмотр форм), в списке которого и выбирается нужная форма (рис. 1.5).

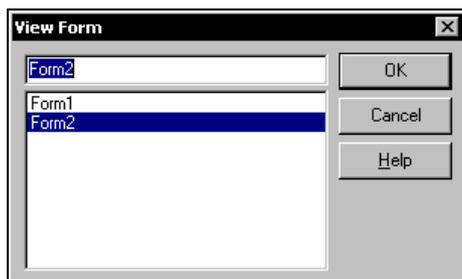


Рис. 1.5. Выбор файла описания формы

Одновременно можно отобразить на экране несколько форм. Для закрытия того или иного окна Формы нужно выполнить команду **File | Close** (Файл | Закрыть) или щелкнуть мышью на кнопке закрытия соответствующего окна.

*Файл модуля формы* содержит описание класса формы. Для пустой формы, добавляемой к проекту по умолчанию, файл модуля формы содержит следующий код:

```
unit Unit1;  
interface
```

```

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms, Dialogs;
type
  TForm1 = class(TForm)
private
  { Private declarations }
public
  { Public declarations }
end;
var
  Form1: TForm1;
implementation
{$R *.dfm}
end.

```

Delphi автоматически создает файл модуля формы при добавлении новой формы. По умолчанию к проекту добавляется новая форма типа `TForm`, не содержащая компонентов.

В разделе `interface` модуля формы содержится описание класса формы, а в разделе `implementation` — подключение к модулю директивой `$R` визуального описания соответствующей формы. При размещении в форме компонентов, а также при создании обработчиков событий в модуль формы вносятся соответствующие изменения. При этом часть этих изменений Delphi выполняет автоматически, а часть пишет разработчик. Обычно все действия разработчика, связанные с программированием, выполняются именно в модулях форм.

Тексты файлов модулей форм отображаются и редактируются с помощью Редактора кода. Открыть файл модуля формы можно в стандартном окне открытия файла (команда **File | Open** (Файл | Открыть)) или в диалоговом окне **View Unit** (рис. 1.6), открываемом командой **View | Units** (Просмотр | Модули) или нажатием комбинации клавиш `<Ctrl>+<F12>`. В окне открытия файла модуля формы можно выбрать также файл проекта. После выбора нужного модуля (или проекта) и нажатия кнопки **ОК** его текст появляется на отдельной странице Редактора кода.

Отметим, что оба файла каждой формы (описания и модуля) имеют одинаковые имена, отличные от имени файла проекта, хотя у имени файла проекта и другое расширение.

При компиляции модуля автоматически создается файл с расширением `dcu` (`dpu` — для Linux), который содержит откомпилированный код модуля. Этот файл можно удалить из каталога, в котором находятся все файлы проекта, — Delphi снова создаст этот файл при следующей компиляции модуля или про-

екта. Смысл создания названных файлов в Delphi состоит в том, что практически можно собрать проект, используя только файлы `dcu` без файлов `pas`.

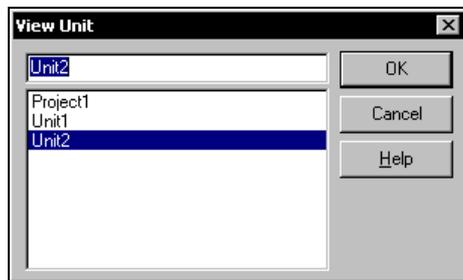


Рис. 1.6. Открытие файла модуля формы

## Файлы модулей

Кроме модулей в составе форм, при программировании можно использовать и *отдельные модули*, не связанные с какой-либо формой. Они оформляются по обычным правилам языка Object Pascal и сохраняются в отдельных файлах. Для подключения модуля его имя указывается в разделе `uses` того модуля или проекта, который использует средства этого модуля.

В отдельном модуле можно (даже полезно) размещать процедуры, функции, константы и переменные, общие для нескольких модулей проекта.

## Файл ресурсов

При первом сохранении проекта автоматически создается файл ресурсов (расширение `res`) с именем, совпадающим с именем файла проекта. Файл ресурсов может содержать следующие ресурсы:

- значки;
- растровые изображения;
- курсоры.

Перечисленные компоненты являются ресурсами Windows, поскольку они разработаны и интерпретируются в соответствии со стандартами этой операционной системы.

Первоначально файл ресурсов содержит значок проекта, которым по умолчанию является изображение факела. В дальнейшем его можно изменить или заменить.

Для работы с файлами ресурсов в состав Delphi включен графический редактор Image Editor версии 3.0, вызываемый командой **Tools | Image Editor**

(Средства | Редактор изображений). На рис. 1.7 показан вид окна Редактора изображений с загруженным файлом ресурсов Project1.res для редактирования значка приложения.

Файл ресурсов имеет иерархическую структуру, в которой ресурсы разбиты на группы, и каждый ресурс имеет уникальное в пределах группы имя. Имя ресурса задается при его создании и в последующем используется в приложении для доступа к этому ресурсу. Значок проекта находится в группе **Icon** и по умолчанию имеет имя **MAINICON**.

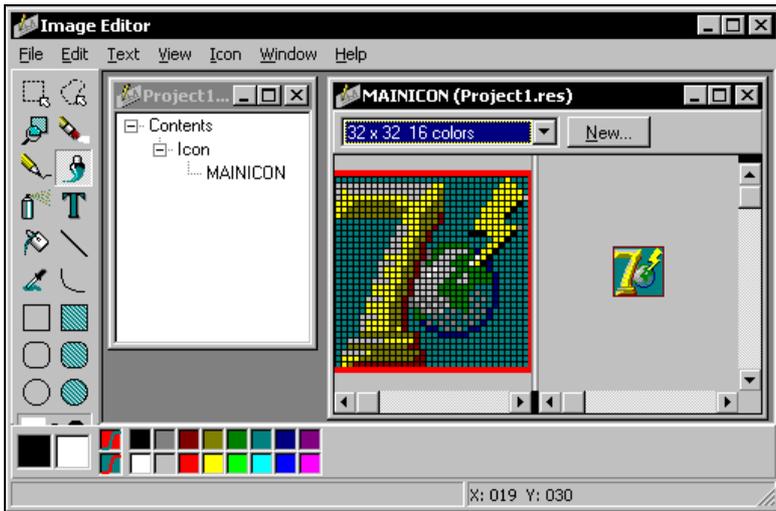


Рис. 1.7. Окно Редактора изображений

Кроме файла с расширением `res`, объединяющего несколько ресурсов, редактор Image Editor также позволяет работать с файлами, содержащими следующие ресурсы (в скобках указано расширение имени файла):

- значки компонентов (`dcr`);
- растровые изображения (`bmp`);
- значки приложений (`ico`);
- курсоры (`cur`).

## Параметры проекта

Для установки параметров проекта используется окно параметров проекта (**Project Options**), открываемое командой **Project | Options** (Проект | Параметры) или нажатием комбинации клавиш `<Shift>+<Ctrl>+<F11>`. Параметры

разбиты на группы, каждая из которых располагается в окне параметров проекта на своей странице (рис. 1.8).

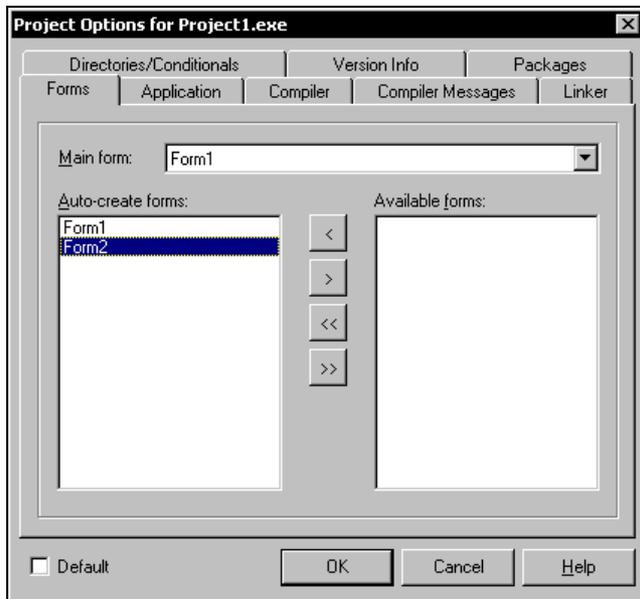


Рис. 1.8. Окно параметров проекта

После установки отдельных параметров Delphi автоматически вносит нужные изменения в соответствующие файлы проекта. Так, параметры из страниц **Forms** и **Application** вносятся в файлы проекта и ресурсов, а параметры из страниц **Compiler** и **Linker** — в файл параметров проекта.

Далее для примера приводятся фрагменты файла параметров проекта.

```
[Compiler]
A=8
B=0
C=1
...
[Version Info Keys]
CompanyName=
FileDescription=
FileVersion=1.0.0.0
...
```

Как видите, файл параметров проекта представляет собой текстовый файл, в котором построчно записаны параметры и их значения.