



Самоучитель

+ ДИСТРИБУТИВ CodeGear RAD Studio 2007

Никита Культин

Основы программирования в Delphi 2007



Среда разработки

Назначение базовых компонентов

Программирование графики,
мультимедиа и баз данных

Новая СУБД Borland Blackfish SQL

Создание справочной системы
и установочного CD

Никита Культин

**Основы программирования
в Delphi 2007**

Санкт-Петербург

«БХВ-Петербург»

2008

УДК 681.3.06
ББК 32.973.26-018.2
К90

Культин Н. Б.

К90 Основы программирования в Delphi 2007. — СПб.:
БХВ-Петербург, 2008. — 480 с.: ил. + DVD — (Самоучитель)

ISBN 978-5-9775-0235-1

Книга является пособием для начинающих по разработке Win32-приложений в Delphi 2007. В ней в доступной форме изложены принципы визуального проектирования и событийного программирования, на конкретных примерах показана методика создания программ различного назначения, приведено описание среды разработки и базовых компонентов. Рассмотрены вопросы программирования графики, мультимедиа, разработки программ работы с базами данных Microsoft Access и Borland Blackfish SQL. Многочисленные примеры демонстрируют назначение компонентов, раскрывают тонкости программирования в Delphi. В приложении приведено описание базовых компонентов и наиболее часто используемых функций. Книга отличается доступностью изложения, большим количеством примеров и адресована студентам, школьникам старших классов и всем, кто хочет научиться программировать.

Прилагаемый DVD содержит проекты, приведенные в книге, а также демо-версию CodeGear RAD Studio 2007, в состав которой входит Delphi 2007.

Для начинающих программистов

УДК 681.3.06
ББК 32.973.26-018.2

Группа подготовки издания:

Главный редактор	<i>Екатерина Кондукова</i>
Зам. главного редактора	<i>Игорь Шишигин</i>
Зав. редакцией	<i>Григорий Добин</i>
Редактор	<i>Екатерина Капальгина</i>
Компьютерная верстка	<i>Натальи Смирновой</i>
Корректор	<i>Зинаида Дмитриева</i>
Дизайн обложки	<i>Инны Тачиной</i>
Зав. производством	<i>Николай Тверских</i>

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 31.01.08.

Формат 70×100^{1/16}. Печать офсетная. Усл. печ. л. 38,7.

Тираж 2000 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 978-5-9775-0235-1

© Культин Н. Б., 2008
© Оформление, издательство "БХВ-Петербург", 2008

Оглавление

ПРЕДИСЛОВИЕ	1
CodeGear Delphi 2007 — что это?	1
Об этой книге	2
ЧАСТЬ 1. DELPHI 2007	3
ГЛАВА 1. СРЕДА РАЗРАБОТКИ DELPHI 2007	5
Установка	5
Первое знакомство	6
ГЛАВА 2. ПЕРВЫЙ ПРОЕКТ	11
Начало работы	11
Форма	12
Компоненты	18
Событие и процедура обработки события	30
Язык программирования Delphi	35
Редактор кода	35
Система подсказок	36
Шаблоны кода	37
Справочная информация	38
Сохранение проекта	38
Структура проекта	40
Компиляция	43
Ошибки	44
Предупреждения и подсказки	46
Запуск программы	46
Исключения	47
Обработка исключения	48
Внесение изменений	53
Настройка приложения	56
Установка приложения на другой компьютер	59

ГЛАВА 3. КОМПОНЕНТЫ	60
Базовые компоненты	60
<i>Label</i>	60
<i>Edit</i>	64
<i>Button</i>	67
<i>CheckBox</i>	70
<i>RadioButton</i>	73
<i>ComboBox</i>	75
<i>ListBox</i>	80
<i>Memo</i>	84
<i>Timer</i>	86
<i>Panel</i>	89
<i>ControlBar</i>	91
<i>SpeedButton</i>	92
<i>StatusBar</i>	97
<i>UpDown</i>	100
<i>ProgressBar</i>	104
<i>Image</i>	107
<i>MainMenu</i>	113
<i>OpenDialog</i>	119
<i>SaveDialog</i>	121
Компоненты Vista	124
<i>TaskDialog</i>	124
<i>FileOpenDialog</i> и <i>FileSaveDialog</i>	132
ЧАСТЬ 2. ПРАКТИКУМ ПРОГРАММИРОВАНИЯ	141
ГЛАВА 4. ГРАФИКА	143
Графическая поверхность	143
Карандаш и кисть.....	146
Графические примитивы.....	148
Текст.....	150
Линия.....	155
Ломаная линия	160
Прямоугольник.....	161
Многоугольник (полигон).....	166
Окружность и эллипс.....	168
Дуга	168
Сектор	170
Точка	175

Битовые образы.....	176
Мультипликация	181
Движение	181
Взаимодействие с пользователем.....	186
Использование битовых образов.....	192
Загрузка битового образа из ресурса	199
Создание файла ресурсов	199
Подключение файла ресурсов.....	201
Загрузка битового образа	202
ГЛАВА 5. МУЛЬТИМЕДИА	204
Функция <i>PlaySound</i>	204
Компонент <i>MediaPlayer</i>	205
Воспроизведение MIDI-звука	209
Проигрыватель CD.....	213
Просмотр видеороликов.....	222
Компонент <i>Animate</i>	229
ГЛАВА 6. БАЗЫ ДАННЫХ	233
База данных и СУБД	233
Локальные и удаленные базы данных	233
Структура базы данных.....	234
Механизмы доступа к данным.....	235
Компоненты доступа к данным	236
Создание базы данных	236
База данных Microsoft Access	237
Доступ к данным	237
Отображение данных.....	243
Выбор информации из базы данных	248
SQL-запрос.....	249
Фильтр.....	254
Работа с базой данных в режиме формы	256
Загрузка строки соединения из INI-файла	265
База данных Blackfish SQL	266
Установка.....	267
Доступ к серверу	268
Создание базы данных	268
Доступ к базе данных	273
Права пользователей	273
База данных <i>Книги</i>	276

Развертывание приложения работы с базой данных	281
Установка и настройка Blackfish SQL	281
Установка программы работы с базой данных	283
ГЛАВА 7. КОМПОНЕНТ ПРОГРАММИСТА.....	286
Модуль компонента	287
Тестирование модуля компонента	296
Установка компонента	300
Ресурсы компонента	301
Создание пакета	303
Установка пакета	304
Тестирование компонента.....	305
Установка компонента на другой компьютер.....	309
Распространение компонента.....	310
ГЛАВА 8. СПРАВОЧНАЯ ИНФОРМАЦИЯ	311
Справочная система HTML Help.....	311
Подготовка справочной информации	312
Microsoft HTML Help Workshop	315
Файл проекта	315
Файл контента (оглавление)	318
Идентификаторы разделов	321
Компиляция	323
Отображение справочной информации	324
ГЛАВА 9. СОЗДАНИЕ УСТАНОВОЧНОГО ДИСКА	328
Утилита InstallAware.....	328
Новый проект	329
Общая информация	331
Программа и ее разработчик.....	331
Требования к системе	333
Компоненты.....	333
Архитектура	334
Возможности	334
Файлы.....	335
Ярлыки	336
Интерфейс.....	338
Диалоги	338
Информация о программе и лицензионное соглашение.....	340
Образ установочного диска	341

ГЛАВА 10. ПРИМЕРЫ ПРОГРАММ.....	344
Экзаменатор	344
Требования к программе	345
Файл теста.....	345
Форма приложения	348
Отображение иллюстрации.....	350
Доступ к файлу теста	351
Текст программы	352
Запуск программы.....	364
Сапер	366
Правила и представление данных	367
Форма	369
Игровое поле	370
Начало игры.....	370
Игра	375
Справочная информация	380
Информация о программе	381
Текст программы	384
MP3-плеер.....	396
Форма	397
Регулятор громкости.....	400
Перемещение окна.....	402
Листинг	403
ПРИЛОЖЕНИЯ	411
ПРИЛОЖЕНИЕ 1. СПРАВОЧНИК.....	413
Форма.....	413
Базовые компоненты	415
<i>Label</i>	415
<i>Edit</i>	416
<i>Button</i>	417
<i>Memo</i>	418
<i>RadioButton</i>	419
<i>CheckBox</i>	420
<i>ListBox</i>	421
<i>ComboBox</i>	422
<i>StringGrid</i>	423
<i>Image</i>	425

<i>Timer</i>	426
<i>SpeedButton</i>	427
<i>UpDown</i>	429
<i>OpenDialog</i>	430
<i>SaveDialog</i>	431
<i>Animate</i>	432
<i>MediaPlayer</i>	433
Компоненты доступа-манипулирования данными	434
<i>ADoConnection</i>	434
<i>ADoTable</i>	435
<i>ADoDataSet</i>	436
<i>ADoQuery</i>	437
<i>DataSource</i>	438
<i>DBEdit, DBMemo, DBText</i>	439
<i>DBGrid</i>	439
<i>DBNavigator</i>	441
Графика.....	443
<i>PaintBox</i>	443
<i>Canvas</i>	443
<i>Pen</i>	446
<i>Brush</i>	446
Цвет	447
Функции.....	448
Функции ввода и вывода.....	448
Математические функции	449
Функции преобразования.....	449
Функции манипулирования датами и временем.....	450
События	452
Исключения.....	453

ПРИЛОЖЕНИЕ 2. СОДЕРЖИМОЕ DVD	455
---	------------

РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА	461
---------------------------------------	------------

ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ	462
-----------------------------------	------------

Предисловие

CodeGear Delphi 2007 — что это?

Среда разработки Borland Delphi является одним из популярнейших инструментов разработки прикладных программ (приложений). Она ориентирована на так называемую "быструю" разработку, в основе которой лежит технология визуального проектирования и событийного программирования, суть которой состоит в том, что среда разработки берет на себя большую часть рутинной работы, оставляя программисту работу по созданию диалоговых окон и процедур обработки событий. Производительность программиста при этом просто фантастическая! Следует обратить внимание на то, что хотя среда разработки и язык программирования являются объектно-ориентированными, тем не менее для работы в ней не требуется фундаментальных знаний в области объектно-ориентированного программирования, достаточно понимания концепции.

Delphi — это среда быстрой разработки приложений (RAD-среда, от Rapid Application Development — быстрая разработка приложений) на языке Delphi, в основе которого лежит хорошо знакомый многим программистам язык Pascal.

Изначально, вплоть до седьмой версии, Delphi была ориентирована на разработку приложений для Windows. После того как Microsoft стала продвигать технологию .NET, фирма Borland выпустила Delphi 8 for The Microsoft .NET Framework — среду разработки .NET-приложений. Следующая версия Delphi, ориентированная на разработку Windows-приложений, получила название Borland Delphi 2005 for Win32, а среда разработки .NET-приложений — Borland Delphi 2005 for .NET.

Теперь, после реорганизации Borland Corporation и создания в ее структуре нового подразделения CodeGear, разработчикам стала доступной очередная версия Delphi — CodeGear Delphi 2007 for Win32.

CodeGear Delphi 2007 for Win32 существует в двух вариантах: Professional и Enterprise. Каждый комплект включает набор средств и компонентов, обеспечивающих разработку высокоэффективных приложений различного назначения. Возможности Enterprise несколько больше, так, например, в него включены драйверы доступа к базам данных Oracle, MS SQL Server, Informix, которых нет в версии Professional. Помимо среды разработки, в комплект по-

ставки Delphi включен новый сервер баз данных CodeGear BlackFish SQL Server.

Следует обратить внимание, Delphi 2007 for Win32 доступна как отдельный инструмент разработки, а также как элемент CodeGear RAD Studio 2007. Более подробную информацию о продуктах CodeGear можно найти на сайте фирмы (<http://codegear.com>).

CodeGear Delphi 2007 for Win32 может работать в среде операционных систем Microsoft Windows 2000, Microsoft Windows XP Professional, Microsoft Windows Server 2003, а также в Microsoft Windows Vista. Особых требований, по современным меркам, к ресурсам компьютера среда разработки не предъявляет: процессор должен быть класса Pentium III 1,4 ГГц или более мощный (рекомендуется Pentium III с частотой 850+ МГц), 512 Мбайт оперативной памяти (рекомендуется более 1 Гбайт), 750 Мбайт свободного места на жестком диске (в том числе для Microsoft .NET Framework и Microsoft .NET SDK).

Об этой книге

Книга, которую вы держите в руках, — это не описание среды разработки или языка программирования, это руководство, пособие по программированию в CodeGear Delphi 2007 for Win32, разработке Win32-приложений. Здесь представлена технология визуального проектирования и событийного программирования, на конкретных примерах рассмотрен процесс создания программы: от разработки диалогового окна и процедур обработки событий до создания справочной системы и образа установочного диска.

Цель книги — научить программировать, создавать программы различного назначения: от простых однооконных приложений до программ работы с базами данных. Следует обратить внимание, что хотя книга ориентирована на читателя, обладающего определенными знаниями и начальным опытом в программировании, она вполне доступна и начинающим.

Научиться программировать можно только программируя, решая конкретные задачи. Поэтому, чтобы получить максимальную пользу от книги, вы должны работать с ней активно. Изучайте листинги, старайтесь понять, как работают программы. Не бойтесь экспериментировать — совершенствуйте программы, вносите в них изменения. Чем больше вы сделаете самостоятельно, тем большему вы научитесь!



ЧАСТЬ 1

DELPHI 2007

В первой части книги приведено краткое описание среды разработки Delphi 2007; на примере программы Конвертор показан процесс разработки приложения; приведено описание базовых компонентов.

ГЛАВА 1



Среда разработки Delphi 2007

Установка

CodeGear предоставляет два варианта приобретения своих продуктов: "коробка" и установка с сервера (Download). В первом случае пользователь получает DVD, на котором находятся все необходимые файлы, и серийный номер. При покупке варианта Download пользователь получает серийный номер и ссылку на программу активизации установки, а все необходимые для установки файлы загружаются с сервера CodeGear.

Процесс установки с DVD активизируется автоматически (запускается файл setup.exe), как только диск будет вставлен в дисковод. Для того чтобы установить Delphi с сервера CodeGear, надо запустить программу активизации установки (файл CodeGearRADStudio2007Setup.exe, загруженный с сервера на первом этапе).

Delphi 2007 является .NET-приложением. Поэтому установка начинается с проверки наличия на компьютере разработчика Microsoft .NET Framework 2.0, Microsoft .NET Framework SDK 2.0 и Microsoft Visual J# .NET. Если какой-либо из перечисленных компонентов отсутствует, то он устанавливается. После этого начинается процесс установки Delphi.

Сначала на экране появляется окно лицензионного соглашения, затем окно **Select Features**, в нем перечислены компоненты, которые по умолчанию будут установлены на компьютер. Если какой-либо из компонентов программисту не нужен, то он может отказаться от его установки. Но если на диске компьютера достаточно свободного места, то лучше все-таки установить все компоненты.

Первое знакомство

Чтобы запустить Delphi 2007, надо сделать щелчок на кнопке **Пуск** и в меню **Программы** выбрать команду **CodeGear RAD Studio ▶ Delphi 2007**.

Для того чтобы начать работу над новым *проектом*, надо в меню **File** выбрать команду **New ▶ VCL Forms Application - Delphi for Win32**.

Окно **Delphi** в начале работы над новым проектом приведено на рис. 1.1. В верхней части окна находится строка меню и панель инструментов.

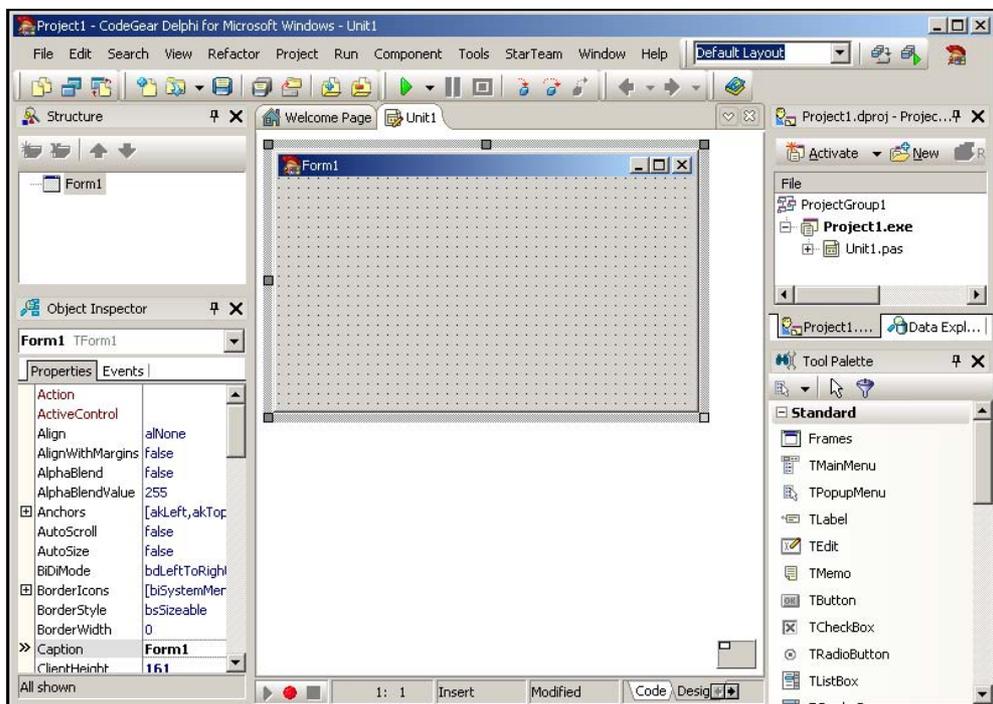


Рис. 1.1. Окно Delphi
в начале работы над новым проектом

Центральную часть окна занимает окно дизайнера формы (рис. 1.2). В нем находится форма — заготовка окна приложения (окно программы во время его разработки принято называть формой).

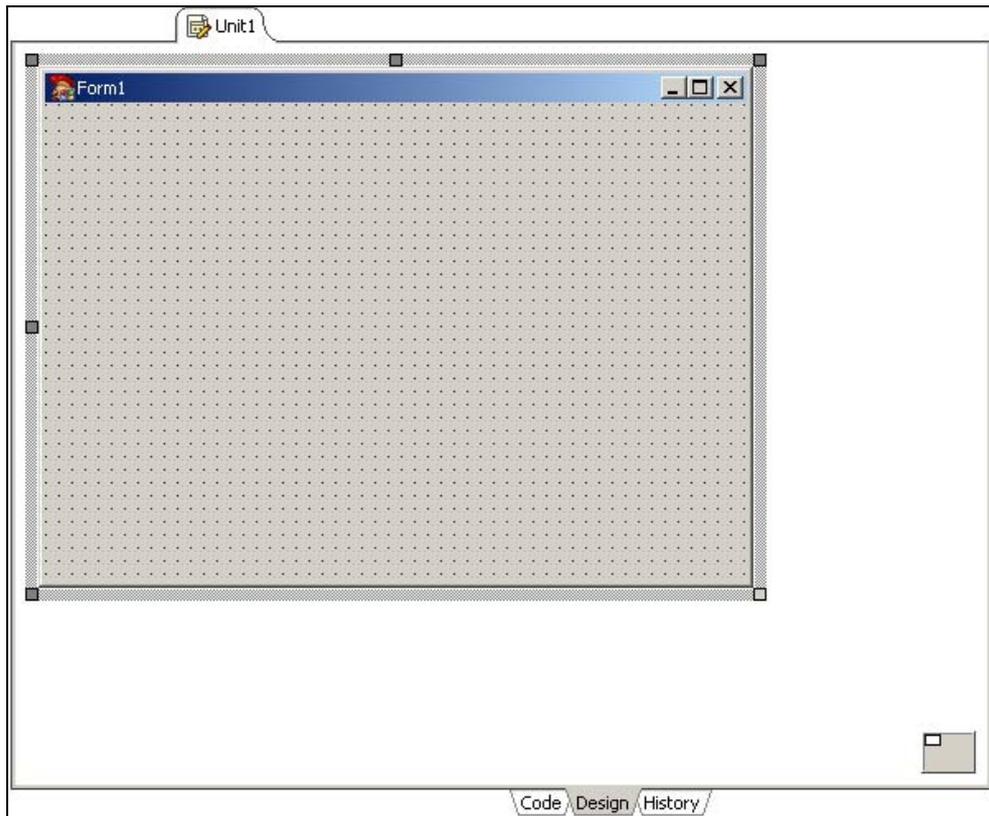


Рис. 1.2. Окно дизайнера формы

За окном дизайнера формы находится окно редактора кода (рис. 1.3), доступ к которому можно получить, сделав щелчок на ярлычке **Code** или нажав клавишу <F12>.

Слева от окна дизайнера формы находится окно **Object Inspector** (рис. 1.4). Вкладка **Properties** используется для редактирования значений *свойств объектов*. Свойство (property) — это характеристика *объекта* (формы, командной кнопки, поля редактирования и т. д.). Свойства определяют вид объекта, его положение и поведение. Например, свойство *Caption* формы определяет текст, который отображается в ее заголовке, а свойства *Width* и *Height* — ее размеры (ширину и высоту). Справа от названия свойств указаны их значения. Свойства по функциональному признаку объединены в группы (названия групп выделены цветом). Так, например, свойства, определяющие внешний вид объекта, объединены в группу **Visual**. Программист может изменить

способ отображения свойств, выбрав в контекстном меню вкладки **Properties** команду **Arrange ▶ by Name** (в алфавитном порядке) или **Arrange ▶ by Category** (по категориям). На вкладке **Events** перечислены события, которые может воспринимать объект.

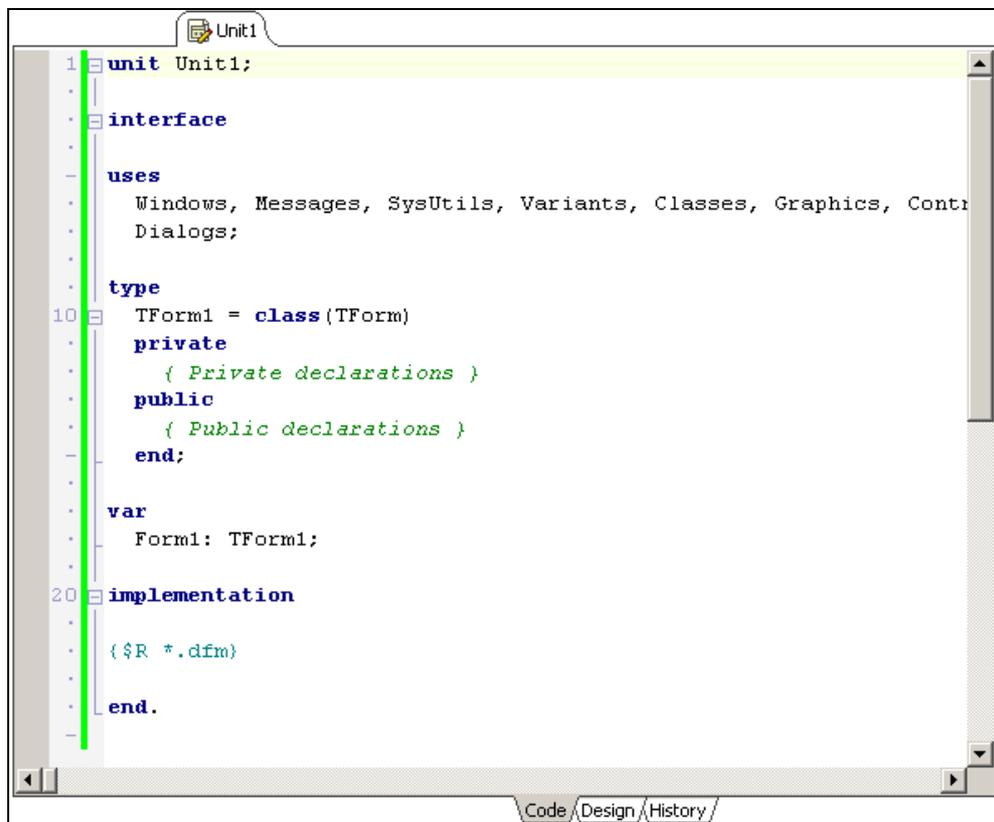


Рис. 1.3. Окно редактора кода

На вкладках окна **Tool Palette** (рис. 1.5) находятся *компоненты*. Компонент — это элемент пользовательского интерфейса или объект, реализующий некоторую функциональность. Например, на вкладке **Standard** находятся компоненты, обеспечивающие взаимодействие с пользователем (Label — поле отображения текста; Edit — поле редактирования; Button — командная кнопка и др.), а на вкладке **dbGo** — компоненты, обеспечивающие доступ к базам данных.

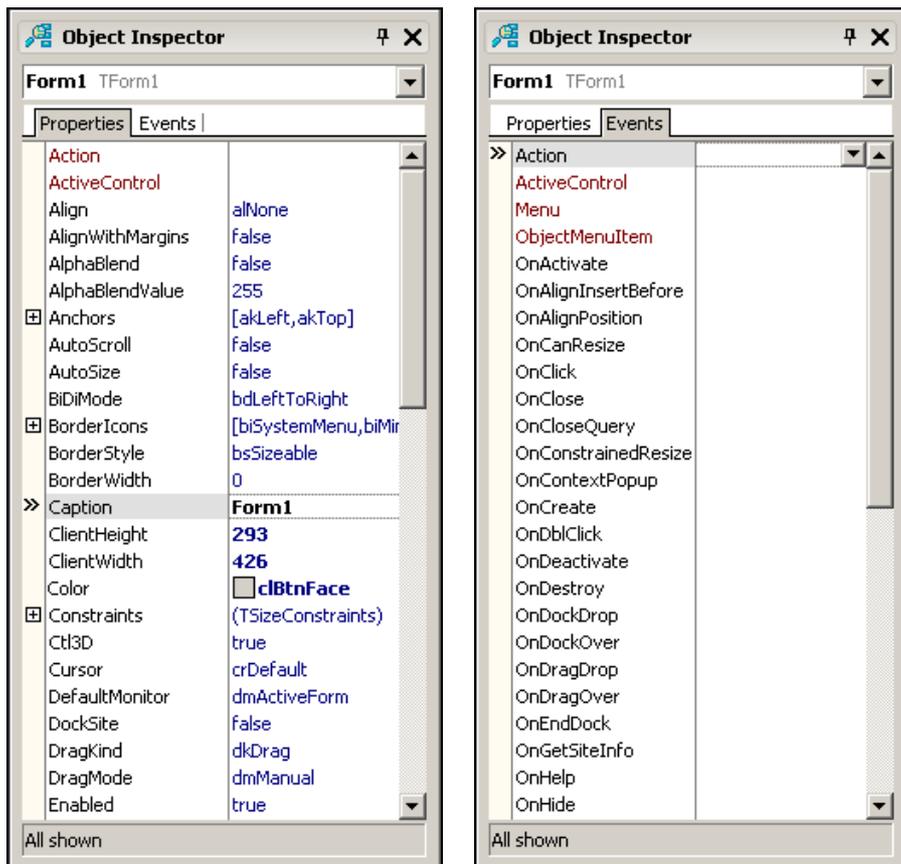


Рис. 1.4. В окне **Object Inspector** на вкладке **Properties** перечислены свойства объекта, а на вкладке **Events** — события, на которые объект может реагировать

ЗАМЕЧАНИЕ

Компоненты, обеспечивающие взаимодействие с пользователем, образуют так называемую VCL-библиотеку (Visual Components Library). Поэтому приложения, использующие эти компоненты, называются VCL-приложениями (VCL Forms Application).

В окне **Project Manager** (рис. 1.6) отображается структура проекта (приложения, над которым в данный момент идет работа).

Если какое-либо из перечисленных окон на экране не отображается, то для того чтобы его увидеть, надо в меню **View** выбрать соответствующую команду.

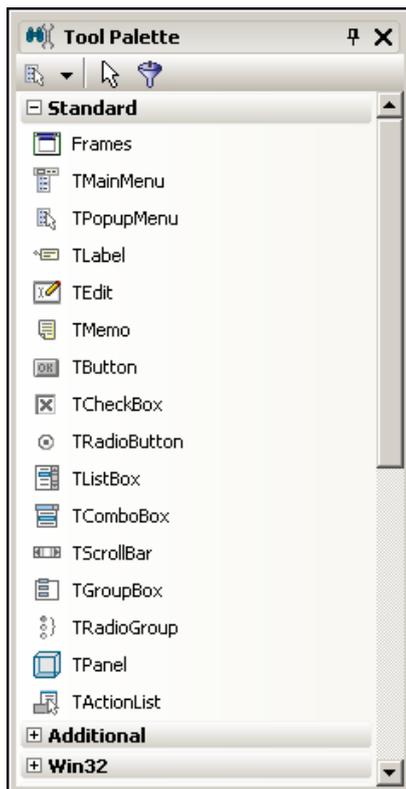


Рис. 1.5. Вкладка **Standard** содержит компоненты, обеспечивающие взаимодействие пользователя с программой

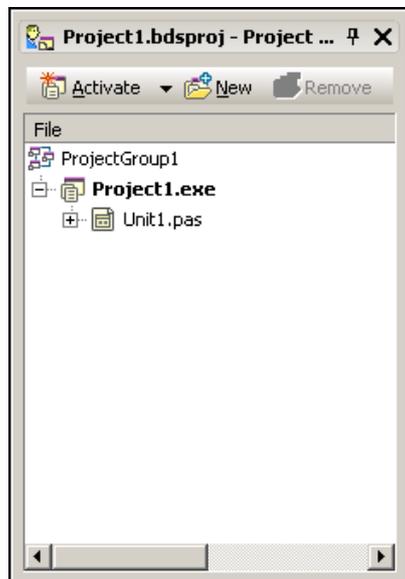


Рис. 1.6. В **Project Manager** отображается структура проекта

ГЛАВА 2



Первый проект

Процесс разработки программы в Delphi рассмотрим на примере — создадим *приложение* (так принято называть прикладную программу), с помощью которого можно пересчитать цену из долларов в рубли. Окно программы приведено на рис. 2.1.

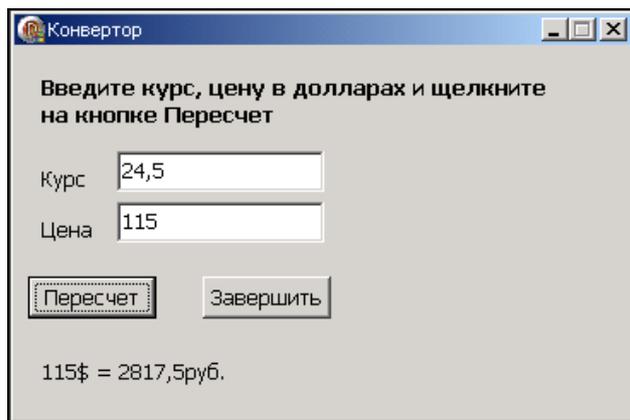


Рис. 2.1. Окно программы Конвертор

Начало работы

Чтобы начать работу над новым проектом (приложением), нужно в меню **File** выбрать команду **New** ▶ **VCL Forms Application - Delphi for Win32**.

Форма

Работа над приложением начинается с создания стартовой *формы* — главного окна программы.

Сначала нужно настроить форму, затем — добавить к форме необходимые *компоненты* (поля отображения информации, поля ввода, командные кнопки).

Настройка формы (компонента) осуществляется путем изменения значений *свойств*. Свойства объекта (формы, компонента) определяют его вид и поведение. Например, свойство `Caption` определяет текст в заголовке формы, а свойство `Position` — положение окна в момент первого его появления на экране. Основные свойства формы приведены в табл. 2.1.

Таблица 2.1. Свойства формы (объекта `TForm`)

Свойство	Описание
Name	Имя (идентификатор) формы. Используется для доступа к форме, ее свойствам и методам, а также для доступа к компонентам формы
Caption	Текст заголовка
Width	Ширина формы
Height	Высота формы
Position	Положение окна в момент первого его появления на экране (<code>poCenterScreen</code> — в центре экрана; <code>poOwnerFormCenter</code> — в центре родительского окна; <code>poDesigned</code> — положение окна определяют значения свойств <code>Top</code> и <code>Left</code>)
Top	Расстояние от верхней границы формы до верхней границы экрана
Left	Расстояние от левой границы формы до левой границы экрана
BorderStyle	Вид границы. Граница может быть обычной (<code>bsSizeable</code>), тонкой (<code>bsSingle</code>) или отсутствовать (<code>bsNone</code>). Если у окна обычная граница, то во время работы программы пользователь может с помощью мыши изменить размер окна. Изменить размер окна с тонкой границей нельзя. Если граница отсутствует, то на экран во время работы программы будет выведено окно без заголовка. Положение и размер такого окна во время работы программы изменить нельзя

Таблица 2.1 (окончание)

Свойство	Описание
BorderIcons	Кнопки управления окном. Значение свойства определяет, какие кнопки управления окном будут доступны пользователю во время работы программы. Значение свойства задается путем присвоения значений уточняющим свойствам <code>biSystemMenu</code> , <code>biMinimize</code> , <code>biMaximize</code> и <code>biHelp</code> . Свойство <code>biSystemMenu</code> определяет доступность кнопки системного меню (значок в заголовке окна), <code>biMinimize</code> — кнопки Свернуть , <code>biMaximize</code> — кнопки Развернуть , <code>biHelp</code> — кнопки вывода справочной информации
Icon	Значок в заголовке диалогового окна, обозначающий кнопку вывода системного меню
Color	Цвет фона. Цвет можно задать, указав название цвета или привязку к текущей цветовой схеме операционной системы. Во втором случае цвет определяется текущей цветовой схемой, выбранным компонентом привязки и меняется при изменении цветовой схемы операционной системы
Font	Шрифт. Шрифт, используемый "по умолчанию" компонентами, находящимися на поверхности формы. Изменение свойства <code>Font</code> формы приводит к автоматическому изменению свойства <code>Font</code> компонента, располагающегося на поверхности формы. То есть компоненты наследуют свойство <code>Font</code> от формы (имеется возможность запретить наследование)

Для изменения значений свойств объектов используется вкладка **Properties** окна **Object Inspector**. В левой колонке этой вкладки перечислены свойства объекта, *выбранного* в данный момент, в правой — указаны значения свойств. Имя выбранного объекта отображается в верхней части окна **Object Inspector**.

На вкладке **Properties** свойства объединены в группы по функциональному признаку (названия групп выделены цветом). Например, группа **Visual** содержит свойства, определяющие вид объекта (для формы — заголовков, цвет фона, вид границы), а группа **Layout** — свойства, определяющие положение объекта (для формы — координаты левого верхнего угла). Некоторые свойства, например `Width` и `Height`, отображаются в нескольких группах (**Visual** и **Layout**).

Программист может изменить способ отображения свойств в окне **Object Inspector**. Например, чтобы свойства отображались в алфавитном порядке, в контекстном меню вкладки **Properties** надо выбрать команду **Arrange ▶ by Name**.

Чтобы в заголовке формы вместо `Form1` появилось название программы — текст `Конвертор`, следует изменить значение свойства `Caption`. Чтобы это сделать, надо в окне **Object Inspector** щелкнуть левой кнопкой мыши в строке свойства (в результате будет выделено текущее значение свойства и появится курсор), ввести текст `Конвертор` и нажать клавишу `<Enter>` (рис. 2.2).

Аналогичным образом можно установить значения свойств `Height` и `Width`, которые определяют высоту и ширину формы. Размер формы, а также размер других компонентов, задают в пикселах (точках). Свойствам `Height` и `Width` надо присвоить значения 200 и 330 соответственно.

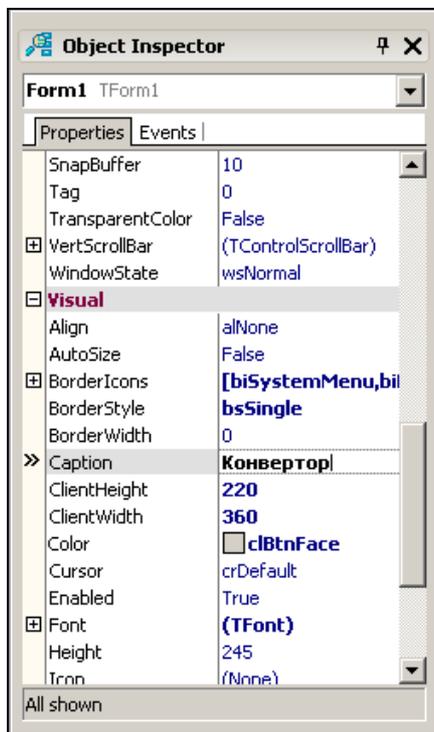


Рис. 2.2. Изменение значения свойства `Caption` путем ввода нового значения

Форма — это окно. Поэтому ее размер можно изменить точно так же, как и любого окна, т. е. путем перемещения границы. По окончании перемещения границы значения свойств `Height` и `Width` будут соответствовать установленному размеру формы.

Положение окна на экране в момент его первого появления соответствует положению формы, заданному во время разработки программы. Положение можно задать, установив значение свойств `Top` (отступ от верхней границы экрана) и `Left` (отступ от левой границы экрана) или задав значение свойства `Position`.

При выборе некоторых свойств, например `BorderStyle`, справа от текущего значения свойства появляется значок раскрывающегося списка. Очевидно, что значение таких свойств можно задать путем выбора из списка (рис. 2.3).

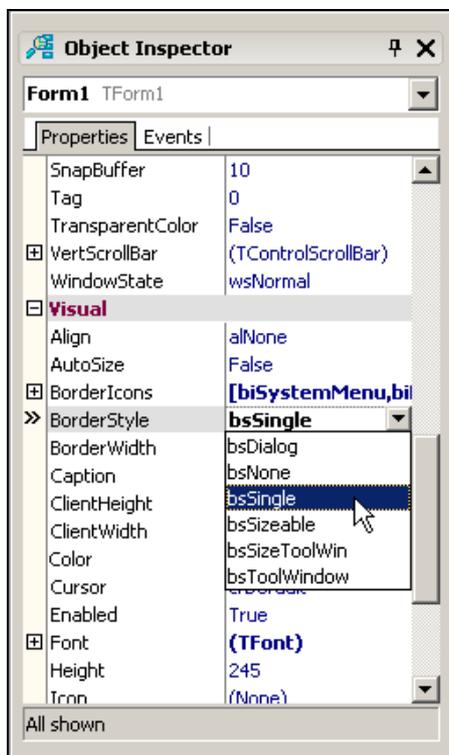


Рис. 2.3. Установка значения свойства путем выбора из списка

Некоторые свойства являются сложными, т. е. их значение определяется совокупностью значений других (уточняющих) свойств. Например, свойство `BorderIcons` определяет кнопки управления окном, которые будут доступны во время работы программы. Значения этого свойства определяются совокупностью значений свойств `biSystemMenu`, `biMinimize`, `biMaximize` и `biHelp`, каждое из которых, в свою очередь, определяет наличие соответствующей командной кнопки в заголовке окна во время работы программы. Перед именами сложных свойств стоит значок "+", в результате щелчка на котором раскрывается список уточняющих свойств (рис. 2.4). Значение уточняющего свойства можно задать обычным образом (ввести значение в поле редактирования или выбрать в списке).

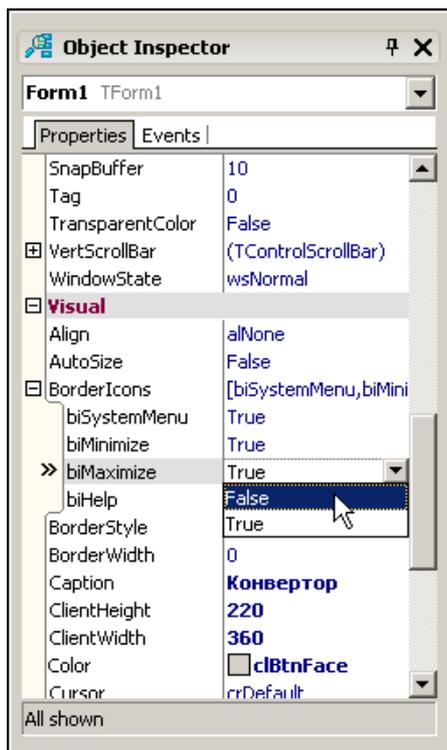


Рис. 2.4. Изменение значения уточняющего свойства

В результате выбора некоторых свойств, например `Font`, в поле значения свойства отображается кнопка, на которой изображены три точки. Это зна-

чит, что задать значение свойства можно в дополнительном диалоговом окне, которое появится в результате щелчка на этой кнопке. Например, значение свойства `Font` можно задать путем ввода значений уточняющих свойств (`Name`, `Size`, `Style` и др.), а можно воспользоваться стандартным диалоговым окном **Шрифт**, которое появится в результате щелчка на кнопке с тремя точками (рис. 2.5).

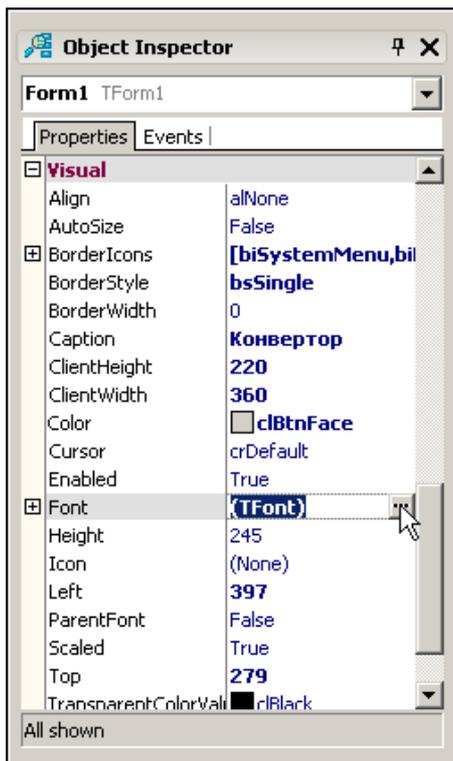


Рис. 2.5. Чтобы задать свойства шрифта, щелкните на кнопке с тремя точками

В табл. 2.2 приведены свойства стартовой формы программы Конвертор, значения которых следует изменить. Значения остальных свойств формы оставлены без изменения, и поэтому в таблице не представлены. В приведенной таблице в именах некоторых свойств есть точка. Это значит, что надо задать значение уточняющего свойства.

Таблица 2.2. Значения свойств стартовой формы

Свойство	Значение	Комментарий
Caption	Конвертор	
Height	220	
Width	360	
BorderStyle	bsSingle	Тонкая граница. Во время работы программы пользователь не сможет изменить размер окна путем захвата и перемещения его границы
BorderIcons.biMaximize	False	В заголовке окна не отображать кнопку Развернуть
Font.Name	Tahoma	
Font.Size	10	

После того как форма будет настроена (будут установлены значения свойств), на форму надо добавить *компоненты*.

Компоненты

Поля редактирования и отображения текста, списки, переключатели, командные кнопки и другие элементы, обеспечивающие взаимодействие программы с пользователем, называют *компонентами пользовательского интерфейса*.

Компоненты, которые программист может использовать в процессе разработки программы, находятся на вкладках палитры компонентов (**Tool Palette**). На вкладках **Standard**, **Additional**, и **Win32** отображаются наиболее часто используемые компоненты пользовательского интерфейса.

Программа пересчета цены из долларов в рубли должна получить от пользователя цену в долларах и курс (соотношение рубля и доллара). Ввод данных с клавиатуры обеспечивает компонент `Edit`. Поэтому на форму разрабатываемого приложения нужно добавить два компонента `Edit`.

Чтобы на форму добавить компонент `Edit`, надо:

1. В палитре компонентов (окно **Tool Palette**) раскрыть вкладку **Standard**.
2. Сделать щелчок на значке компонента (рис. 2.6). Здесь следует обратить внимание, что в палитре компонентов рядом со значком компонента указывается тип, а не название компонента.
3. Установить указатель мыши в ту точку формы, в которой должен быть левый верхний угол компонента.
4. Сделать щелчок левой кнопкой мыши.

В результате на форме появляется поле редактирования — компонент `Edit` (рис. 2.7).

Другие компоненты добавляются на форму аналогичным образом.

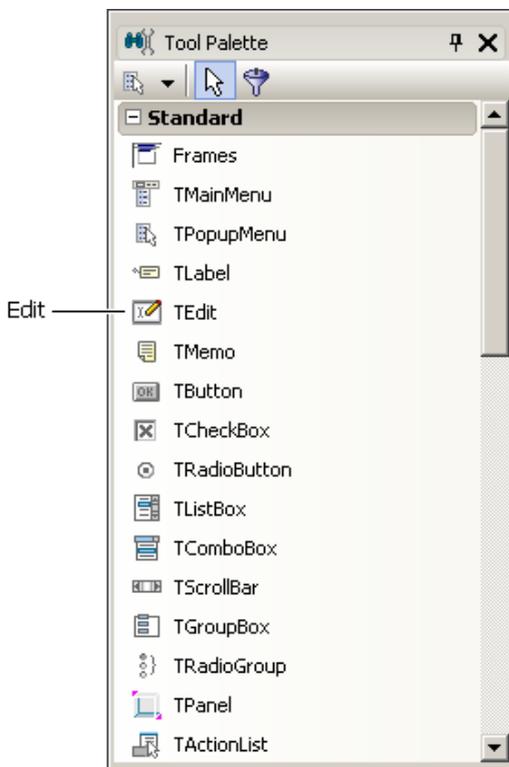


Рис. 2.6. Компонент `Edit` — поле редактирования

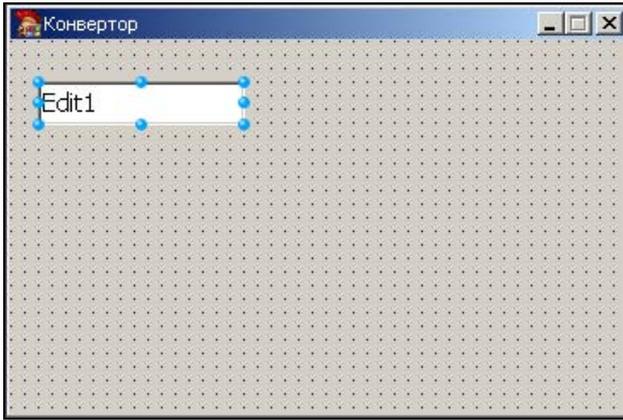


Рис. 2.7. Результат добавления на форму компонента `Edit`

Каждому добавленному компоненту среда разработки присваивает имя, которое состоит из названия компонента и его порядкового номера. Например, первый компонент `Edit` получает имя `Edit1`, второй — `Edit2`. Программист, путем изменения значения свойства `Name`, может изменить имя компонента. Однако в простых программах имена компонентов, как правило, не изменяют.

Основные свойства компонента `Edit` приведены в табл. 2.3.

Таблица 2.3. Свойства компонента `Edit` (объект типа `TEdit`)

Свойство	Описание
<code>Name</code>	Имя компонента. Используется для доступа к компоненту и его свойствам
<code>Text</code>	Текст, который находится в поле редактирования
<code>Left</code>	Расстояние от левой границы компонента до левой границы формы
<code>Top</code>	Расстояние от верхней границы компонента до верхней границы формы
<code>Height</code>	Высота компонента
<code>Width</code>	Ширина компонента
<code>Font</code>	Шрифт, используемый для отображения текста в поле компонента

Таблица 2.3 (окончание)

Свойство	Описание
ParentFont	Признак наследования шрифта от формы. Если значения свойства равно True, то для отображения текста в поле компонента используется шрифт формы
MaxLength	Количество символов, которое можно ввести в поле редактирования. Если значение свойства равно нулю, ограничения на количество символов нет
TabOrder	Определяет порядок перемещения фокуса (курсора) с одного элемента управления на другой в результате нажатия клавиши <Tab>

На рис. 2.8 приведен вид формы после добавления двух полей редактирования. Один из компонентов *выбран* (выделен), окружен восемью маленькими кружками. Свойства выбранного компонента отображаются в окне **Object Inspector**. Чтобы увидеть и, если надо, изменить свойства другого компонента, нужно этот компонент выбрать — щелкнуть левой кнопкой мыши на изображении компонента или выбрать имя компонента в раскрывающемся списке, который находится в верхней части окна **Object Inspector** (рис. 2.9). Компонент, свойства которого надо изменить, можно выбрать и в окне **Structure** (рис. 2.10).

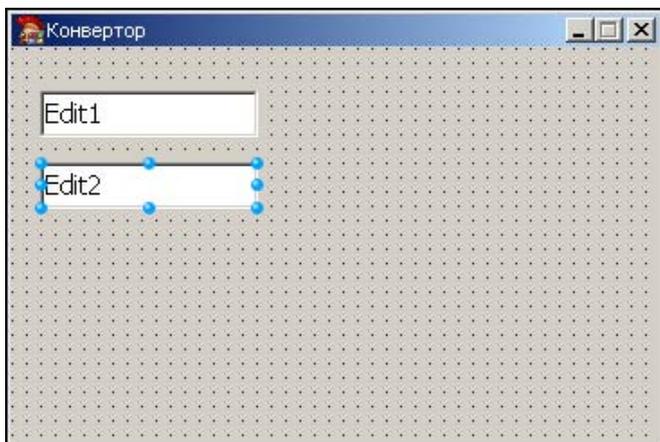


Рис. 2.8. Форма с двумя компонентами