



21世纪高等院校新概念系列教材

Delphi 程序设计基础

Delphi CHENGXU SHEJI JICHU

陈强 黄再忠 主编



中国宇航出版社

Delphi 程序设计基础

陈 强 黄再忠 主编



中国宇航出版社

· 北京 ·

内 容 简 介

本书系统地介绍了 Delphi 程序设计的基本理论与实用编程技术。全书共分为 9 章, 内容包括 Delphi 7 概述、Object Pascal 程序设计语言、面向对象编程、Delphi 7 程序设计基础、窗体和常用组件、多窗体设计与系统对象、程序调试及异常处理、数据库开发和文件管理。

本书注重理论与实践相结合, 注重基础知识的理解与基本技能的培养。书中内容丰富, 章节安排合理, 讲解深入浅出, 示例典型实用, 程序结构严谨规范。

本书可作为高等院校、高职高专院校计算机专业的教材, 也适合于 Delphi 的初学者作为学习和开发 Delphi 应用程序的参考书。

版权所有 侵权必究

图书在版编目 (CIP) 数据

Delphi 程序设计基础/ 陈强, 黄再忠主编. —北京:
中国宇航出版社, 2005. 9

(21 世纪高等院校新概念系列教材)

ISBN 7-80144-987-8

I. D… II. ①陈…②黄… III. 软件工具-程序设计-高等学校-教材 IV. TP311.56

中国版本图书馆 CIP 数据核字 (2005) 第 095928 号

责任编辑 王 雄 封面设计 03 工舍

出 版
发 行

中国宇航出版社

社 址 北京市阜成路 8 号 邮 编 100830
(010) 68768548

网 址 www.caphbook.com / www.caphbook.com.cn

经 销 新华书店

发行部 (010) 68371900 (010) 88530478 (传真)
(010) 68768541 (010) 68767294 (传真)

零售店 读者服务部 北京宇航文苑
(010) 68371105 (010) 62529336

承 印 北京地质印刷厂

版 次 2005 年 9 月第 1 版
2005 年 9 月第 1 次印刷

规 格 787×1092

开 本 1/16

印 张 20.5

字 数 510 千字

书 号 ISBN 7-80144-987-8

定 价 32.00 元

本书如有印装质量问题, 可与发行部调换

前 言

Delphi 7 是 Borland 公司推出的一种基于 Windows 的开发工具。它以可视化的开发环境、简洁高效的编程语言、功能强大的组件、可扩展的数据库等优点，成为目前最流行的编程工具。

Delphi 7 以 Object Pascal 语言为基础，继承了 Pascal 语言的严谨结构和优雅风格，完全地支持面向对象程序设计等诸多特性。

本书是 Delphi 程序设计的入门书，适用于有 Pascal 语言或其他编程语言基础的读者。

本书主要介绍了构成 Delphi 应用程序的多种基本要素和必备知识，包括了 Delphi 的 IDE 可视化开发环境、Object Pascal 语言、Delphi 应用程序的构成、面向对象程序设计的基本概念、窗体和各种功能的组件、多窗体技术、调试技术与异常处理、数据库编程、文件管理。

本书力求使读者能快速、轻松地学会用 Delphi 编程，通过大量的实例进行讲解。相信初学者可以通过本书循序渐进的实例和技巧学习，对 Delphi 编程产生浓厚的兴趣。书中所有的程序实例都是在 Delphi 7+ Windows 2000 环境下调试通过的。

本书由陈强、黄再忠担任主编，刘福道、胡云辉、刘丽君担任副主编，参加编写的有：陈铿、庄金锋、张朋、刘肃平、陈自红、潘闪闪。全书由陈强制订编写大纲并负责统稿工作。

由于作者水平有限，书中难免有疏漏之处，恳请广大读者批评指出。

E-mail: cqjxnc@sina.com.cn

编者

2005 年 6 月

目 录

第 1 章 Delphi 7 概述

- 1.1 Delphi 7 的新特性 (1)
- 1.2 Delphi 7 的集成开发环境 (3)
 - 1.2.1 主窗体 (4)
 - 1.2.2 窗体 (5)
 - 1.2.3 对象树浏览器 (6)
 - 1.2.4 对象监视器 (6)
 - 1.2.5 代码编辑器 (7)
 - 1.2.6 Delphi 7 的系统功能和帮助 (7)

- 习题 1 (9)

第 2 章 Object Pascal 程序设计语言

- 2.1 控制台程序 (10)
- 2.2 注释 (13)
- 2.3 标识符与保留字 (14)
 - 2.3.1 标识符的组成和作用 (14)
 - 2.3.2 保留字和指令字 (14)
- 2.4 基本数据类型 (15)
 - 2.4.1 整型 (15)
 - 2.4.2 实型 (17)
 - 2.4.3 布尔型 (17)
 - 2.4.4 字符型 (17)
 - 2.4.5 枚举类型 (17)
 - 2.4.6 字符串类型 (18)
- 2.5 复杂数据类型 (18)
 - 2.5.1 集合类型 (18)
 - 2.5.2 数组类型 (18)
 - 2.5.3 记录类型 (19)
 - 2.5.4 指针类型 (21)
 - 2.5.5 可变类型 (22)
- 2.6 常量与变量 (23)
 - 2.6.1 常量 (23)

- 2.6.2 变量 (23)
- 2.7 运算符和表达式 (24)
 - 2.7.1 运算符 (24)
 - 2.7.2 表达式 (27)
- 2.8 语句 (28)
 - 2.8.1 语句的格式 (28)
 - 2.8.2 赋值语句 (28)
 - 2.8.3 goto 语句 (29)
 - 2.8.4 复合语句 (29)
 - 2.8.5 with 语句 (30)
 - 2.8.6 条件语句 (30)
 - 2.8.7 循环语句 (31)
 - 2.8.8 break 过程 (33)
 - 2.8.9 continue 过程 (34)
- 2.9 过程与函数 (35)
 - 2.9.1 过程 (36)
 - 2.9.2 函数 (37)
- 2.10 消息框过程和函数 (39)
 - 2.10.1 ShowMessage 过程 (39)
 - 2.10.2 MessageDlg 函数 (39)
- 习题 2 (40)

第 3 章 面向对象编程

- 3.1 对象和类 (42)
 - 3.1.1 类的定义 (43)
 - 3.1.2 数据域 (43)
 - 3.1.3 方法 (44)
 - 3.1.4 属性 (44)
 - 3.1.5 事件 (44)
- 3.2 类的特性 (45)
 - 3.2.1 封装与继承 (45)
 - 3.2.2 构造函数和析构函数 (47)

3.2.3 多态性	(50)	5.5.1 ListBox	(105)
3.2.4 self 关键字	(54)	5.5.2 ComboBox	(107)
习题 3	(55)	5.5.3 TreeView	(111)
第 4 章 Delphi 7 程序设计基础		5.5.4 ListView	(116)
4.1 Delphi 7 的基本构成	(56)	5.5.5 ImageList	(119)
4.1.1 工程文件	(56)	5.5.6 ActionList	(122)
4.1.2 单元文件	(57)	5.5.7 CheckListBox	(125)
4.2 Delphi 7 的可视化应用程序	(58)	5.6 表格组件	(126)
4.2.1 Delphi 7 程序设计流程	(58)	5.6.1 StringGrid	(126)
4.2.2 Delphi 7 的第一个可视化应用程序	(58)	5.6.2 DrawGrid	(128)
习题 4	(62)	5.7 菜单组件	(130)
第 5 章 窗体和常用组件		5.7.1 主菜单	(130)
5.1 窗体和组件	(63)	5.7.2 弹出式菜单	(133)
5.1.1 窗体	(63)	5.7.3 使用菜单模板	(133)
5.1.2 组件	(72)	5.8 工具栏和状态栏组件	(136)
5.1.3 在窗体中使用组件	(73)	5.8.1 ToolBar	(136)
5.2 文本组件	(77)	5.8.2 CoolBar	(138)
5.2.1 Label	(78)	5.8.3 ControlBar	(138)
5.2.2 Edit	(81)	5.8.4 StatusBar	(139)
5.2.3 Memo	(83)	5.9 图形组件	(141)
5.2.4 MaskEdit	(86)	5.9.1 Image	(141)
5.2.5 RichEdit	(88)	5.9.2 Shape	(142)
5.3 按钮组件	(90)	5.9.3 PaintBox	(143)
5.3.1 Button	(90)	5.10 公共对话框组件	(144)
5.3.2 BitBtn	(91)	5.10.1 文件型对话框组件	(144)
5.3.3 SpeedButton	(92)	5.10.2 字体对话框组件	(146)
5.3.4 CheckBox	(96)	5.10.3 颜色对话框组件	(147)
5.3.5 RadioButton	(97)	5.10.4 查找和替换对话框组件	(148)
5.4 分组组件	(97)	5.10.5 打印和打印设置对话框组件	(149)
5.4.1 GroupBox	(97)	5.11 其他组件	(153)
5.4.2 RadioGroup	(97)	5.11.1 Timer	(153)
5.4.3 Panel	(100)	5.11.2 ProgressBar	(154)
5.4.4 ScrollBox	(101)	5.11.3 ScrollBar	(155)
5.4.5 PageControl	(102)	5.11.4 TrackBar	(158)
5.4.6 TabControl	(104)		
5.5 列表组件	(105)		

5.11.5 Splitter	(158)	8.1.1 Delphi 7 数据库系统概述	(197)
5.11.6 DateTimePicker	(159)	8.1.2 BDE 简介	(198)
5.11.7 MonthCalendar	(160)	8.1.3 数据库资源管理器的功能	(199)
5.11.8 Calendar	(161)	8.2 Delphi 7 数据库桌面系统	(202)
习题 5	(161)	8.2.1 Database Desktop 概述	(202)
第 6 章 多窗体设计与系统对象		8.2.2 设定工作目录与私有目录	(203)
6.1 建立多窗体应用程序	(162)	8.2.3 创建别名	(204)
6.1.1 在工程中添加/移除窗体	(162)	8.2.4 创建数据库表	(205)
6.1.2 建立窗体模板	(164)	8.2.5 数据库表格的属性	(207)
6.1.3 设置主窗体	(165)	8.3 BDE 数据集组件	(211)
6.1.4 在一个窗体中打开另一个窗体	(165)	8.3.1 数据集组件的状态	(211)
6.1.5 动态建立窗体	(166)	8.3.2 数据集组件常用的方法	(212)
6.2 创建 MDI 应用程序	(166)	8.3.3 TTable 组件	(214)
6.2.1 利用模板方式建立 MDI 应用程序	(167)	8.3.4 TQuery 组件	(216)
6.2.2 MDI 的属性和方法	(168)	8.3.5 数据集组件的使用方法	(217)
6.2.3 MDI 的菜单设计	(168)	8.4 数据访问组件	(217)
6.3 Screen 变量	(169)	8.4.1 数据访问组件介绍	(217)
6.4 Application 变量	(170)	8.4.2 数据访问组件的使用	(218)
习题 6	(172)	8.5 数据控制组件	(218)
第 7 章 程序调试及异常处理		8.5.1 数据控制组件简介	(218)
7.1 程序调试	(173)	8.5.2 TDbGrid 组件	(219)
7.1.1 程序错误的类型	(173)	8.5.3 TDBnavigator 组件	(220)
7.1.2 设置编译环境	(175)	8.5.4 TDBText 组件	(221)
7.1.3 程序执行的控制	(177)	8.5.5 TDBEdit 组件	(221)
7.1.4 调试器的设置	(182)	8.5.6 TDBMemo 组件	(221)
7.1.5 观察变量、表达式和属性的值	(186)	8.5.7 TDBImage 组件	(221)
7.2 异常处理	(188)	8.5.8 TDBListBox 组件	(221)
7.2.1 异常概述	(188)	8.5.9 TDBComboBox 组件	(221)
7.2.2 Delphi 内建的异常类	(189)	8.5.10 其他组件	(221)
7.2.3 异常处理	(191)	8.6 Delphi 7 中的 SQL 编程	(222)
7.2.4 自定义异常类	(194)	8.6.1 SQL 查询语句简介	(222)
习题 7	(196)		
第 8 章 数据库开发			
8.1 数据库引擎管理器	(197)		

8.6.2 SQL 语言编程中的运用	(222)	9.2 文本文件	(269)
8.6.3 动态 SQL 语句的编程	(223)	9.2.1 文本文件的定义	(269)
8.6.4 SQL 编程实例	(225)	9.2.2 文本文件的操作	(269)
8.7 ADO 编程	(229)	9.2.3 文本文件的应用实例	(271)
8.7.1 基本 ADO 的体系结构	(229)	9.3 类型文件	(273)
8.7.2 ADO 数据集组件介绍	(229)	9.3.1 类型文件的定义	(273)
8.7.3 应用 ADO 开发数据库	(231)	9.3.2 类型文件的操作	(274)
应用系统举例	(231)	9.3.3 类型文件的应用实例	(274)
8.8 Rave 报表设计	(234)	9.4 无类型文件	(279)
8.8.1 概述	(234)	9.4.1 无类型文件的定义	(279)
8.8.2 Rave VCL/CLX 组件	(235)	9.4.2 无类型文件的操作	(279)
8.8.3 Rave Report 集成开发环境	(237)	9.4.3 无类型文件的应用实例	(279)
8.8.4 Rave 报表组件	(237)	9.5 流文件	(283)
8.8.5 使用 Rave Report5.0 向导	(240)	9.5.1 流的概念	(283)
开发数据报表	(240)	9.5.2 TStream 类	(284)
8.9 应用举例	(245)	9.5.3 TFileStream 类	(285)
8.9.1 建立一个基于 BDE 的学	(245)	9.5.4 流文件的应用实例	(286)
生缴费系统	(245)	9.6 INI 文件	(289)
8.9.2 建立一个基于 ADO 的人	(252)	9.6.1 INI 文件的结构和定义	(290)
事管理系统	(252)	9.6.2 INI 文件的操作	(290)
习题 8	(265)	9.6.3 INI 文件的应用实例	(292)
第 9 章 文件管理		9.7 注册表文件	(293)
9.1 文件目录类过程和函数的使用	(266)	9.7.1 注册表的数据类型	(294)
9.1.1 磁盘操作的过程和函数	(266)	9.7.2 注册表的结构	(295)
9.1.2 目录操作的过程和函数	(266)	9.7.3 注册表的修改	(297)
9.1.3 文件操作的过程和函数	(267)	9.7.4 注册表文件的应用实例	(299)
		习题 9	(301)
		附录 Delphi 7 常用函数	(303)

第 1 章 Delphi 7 概述

Delphi 是 Borland 公司开发的可视化软件开发工具，具有简单、高效、功能强大的特点。和 Visual C++ 相比，Delphi 更简单、更易于掌握，而在功能上却丝毫不逊色；与 Visual Basic 相比，Delphi 功能更强大、更实用。可以说 Delphi 同时具备了 Visual C++ 功能强大和 VB 简单易学的特点。它一直是程序员非常喜欢的编程工具。因此，“真正的程序员用 C，聪明的程序员用 Delphi”这句话是对 Delphi 最经典、最恰当的描述。

本章介绍 Delphi 的历史、功能及 IDE 环境。

1.1 Delphi 7 的新特性

Delphi 是第四代编程语言，是 RAD（Rapid Application Development，快速应用程序开发）工具的代表。从核心上说，Delphi 是一个 Pascal 编译器。Delphi 语言的发展历程如下。

从 1995 年的 Delphi 1.0 到 1998 年的 Delphi 4.0 四个版本，Borland 放弃了 OWL，取而代之的是全新的 Visual Component Library，从此开始了组件（Component）技术的时代，通过不断地改进和完善，增加了数据库和客户端/服务器的功能、Code Insight（代码内视）、Module Explore（模块导航）等许多新技术。与 Delphi 以前的版本相比，Delphi 5 是 32 位的应用开发环境。它具有一系列新特性以满足应用程序开发过程中的各种需求。如改进的集成开发环境、强劲的工程管理功能、先进的调试技术、扩展的语言特性以及其强大的数据库应用程序开发能力。

Delphi 6 不但在传统的开发能力方面持续改进，让程序员能够享有更具生产力的开发环境，而且对于最新的信息技术，例如 XML/XSL、SOAP、Web Service 等也都有非常良好的支持，让程序员能够快速地使用这些新技术。此外，Delphi 6 在运行时函数库以及编译器方面也有了不少改善，不但让程序员可以开发功能更强的窗口应用程序，也可以更平顺地移植到 Linux 平台，再搭配跨平台的新可视化组件 Framework CLX，即使是 Windows 程序员也可以快速地开发 Linux 平台上的应用系统。

然而，Delphi 7 之所以这么快地推出，很大原因是由于 Delphi 6 的很多新特性还不是很成熟，在 Delphi 7 中这些新特性得到了很大的改善和提高。下面，我们将介绍 Delphi 7 的一些新特性。

1. 功能更强大的集成开发环境

和以往 Delphi 的每一个新版本一样, Delphi 7 继续改善了开发环境, 帮助程序员提高效率。Delphi 7 在集成开发环境中加入了很多让程序员爱不释手的新功能, 下面是对它们的一个简单介绍。

(1) Code Complete (代码完成)

Code Complete 功能在 Delphi 5 中就已经加入了, 这项功能非常受程序员的欢迎, 因为 Code Complete 可以大幅度地减少程序员需要键入的程序代码, 并且减少键入错误。Delphi 7 的 Code Complete 功能在原有的基础上继续改善, 新的 Code Complete 窗口不但可以由程序员自行调整大小, 而且可以使用不同的颜色代表不同的对象, 例如变量、方法和特性等。Delphi 7 的 Code Complete 窗口加入了色彩分析, 以及对对象分门别类的能力。此外, 新的 Code Complete 窗口也会更聪明地过滤对象能够使用的 PME, 避免了以往 Delphi 5 的 Code Complete 窗口无法显示一些 PME 的困窘。另外, 程序员在新的 Code Complete 窗口中选择使用某个方法之后, Code Complete 会自动地在方法名称之后加上 “ ();” 字符, 把光标停在圆括号之中, 并且自动显示这个方法需要的所有参数, Delphi 的确比以前更聪明、更方便了。

(2) Object TreeView (对象树视图)

另外一个新功能就是 Delphi 7 的 Object TreeView 方式表达组件之间的关系。当程序员在 Object TreeView 窗口中单击了一个组件之后, 这个组件会立刻出现在对象查看器之中, 这时程序员可以改变这个对象的 TreeView 窗口。这个新的窗口除了可以显示窗体中所有的对象之外, 也可以以分层的形式显示各组件之间的逻辑关系。

当窗体复杂, 拥有大量的组件时, Object TreeView 窗口可以让用户迅速找到这个组件, 并且知道与当前组件相关的组件, 非常方便。

此外 Delphi 7 的编辑器也强化了 Class Complete 的功能, 让程序员在定义类别程序代码时拥有更大的弹性, 并且加强了 Interface Complete 的功能, 让程序员在撰写接口程序代码时也能够享有 Code Complete 功能。

(3) 可定制的 View 窗口

Delphi 7 最强劲的集成开发环境功能要算是新的可定制的 View 窗口了。在 Delphi 7 中, 编辑器不但可以让程序员观看和编辑 Object Pascal 的程序代码, 而且如果程序员编辑的是 Web 应用程序, 那么还可以直接在编辑器中查看 HTML 程序代码、脚本语言程序代码, 预览 Web 应用程序执行结果的画面以及 Web 应用程序产生的 HTML 程序代码。

(4) 新添加 Indy Intercepts 和 Indy I/O Handler 构件组

企业版的 Delphi 7 中, 这两个构件组基本上提供了针对现今流行的 Internet 协议编程的构件。可以进行更为底层的操作。

(5) 新添加 IW Standard、IW Data、IW Client Side 和 IW Control 构件组

这些构件也是 Delphi 7 的显著改革之一, 将 IntraWeb 集成到了标准的构件板中, 就可以很方便地开发基于 Web 的应用程序。

(6) 新添加的 Rave 报表构件组以很方便地开发基于 Web 的应用程序

Delphi 7 的标准构件板中已经没有了原来的 QuickReport 构件组, 取而代之的是 Rave 报表制作组件, 并且提供专门的报表制作工具 Rave Designer。

Rave 报表构件组是一个可视化的报表设计工具, 大大简化了在应用程序中加入报表的工

作。用 Rave 报表构件可以做出各种不同的报表，从最简单的到非常复杂、个性化的报表都可以完成。它的功能包括：图片支持、对齐、精确页面定位、打印机设置、字体控制、打印预览、报表内容重用，并支持 PDF、HTML、RTF 等文本格式。

2. 新一代数据存取引擎——DBExpress

Borland 为了让 Windows 平台上的 Delphi 以及 Linux 平台上的 Kylix 拥有共同的数据存取引擎，决定开发新一代的数据存取引擎，这就是 DBExpress。DBExpress 是一组存取各种不同关联数据库的原生驱动程序，以及一组基于这些驱动程序而开发的统一的组件。由于原生驱动程序和组件的开发都考虑了跨平台，因此 Delphi 和 Kylix 都可以使用这组驱动程序和组件。程序员通过统一的组件来存取不同的数据库，以方便开发数据库应用系统。

3. 开发 Internet/Intranet 应用系统的 WebSnap 组件组

Delphi 7 中最重要的新增功能之一是能够让程序员开发 Internet/Intranet 应用系统的新架构和组件，这个新功能称为 WebSnap。Delphi 7 的 WebSnap 允许程序员使用组件和可视化的方式快速开发复杂的 Internet/Intranet 应用系统。WebSnap 的特点是允许程序员直接在这些组件之中加入脚本语言，并且能够直接在 Delphi 的集成开发环境中撰写脚本语言、预览 Web 应用程序的输出结果以及 Web 应用程序产生的 HTML 程序代码。

4. MIDAS 的进化版本——DataSnap

在 Delphi 7 中，MIDAS 被改名为 DataSnap。DataSnap 不但强化了 MIDAS 原有的功能，更加入了许多新的组件，让程序员可以使用它开发出更为强劲的应用系统。此外 DataSnap 也改善了 MIDAS 的执行效率，让使用它的应用程序能够执行得更快。

5. Windows XP 应用

Delphi 7 包含了对 Windows XP 风格支持，让开发者能够创建可以利用 Windows XP 使用者界面风格的应用。

总的来说，Delphi 7 不仅改进和固定这些较新的技术使其更健壮（SOAP 支持和 DataSnap 的出现），而且提供对于较新技术（像 Windows XP 主题或者 UDDI）的支持，但是最重要的是它使一套富有吸引力的第三方工具使用更简单：Rave 报表引擎，IntraWeb web 应用开发技术，和 ModelMaker 设计环境。最后，它通过提供第一个不是瞄准 Intel CPU，而是 .NET CIL 平台的 Pascal/Delphi 的 Borland 编译器开辟了一个崭新的世界。

1.2 Delphi 7 的集成开发环境

Delphi7 是用来创建 Windows 应用程序的一种快速应用开发工具，一种可视化编程环境。我们可以更加直观地使用图形化工具来创建 Windows 的应用程序。它使我们从完全以代码的形式创建可视化应用程序的繁琐且复杂的工作中解脱出来，使我们可以不用掌握太多的编程专业知识就能创建出图形化的高难度应用程序，这一切都得益于 Delphi 的集成开发环境。即 IDE（集成开发环境，Integrated Development Environment）。

Delphi7 的可视化开发环境主要包括五部分：主窗口、窗体、对象监视器、代码编辑器和对象树浏览器，如图 1-1 所示。

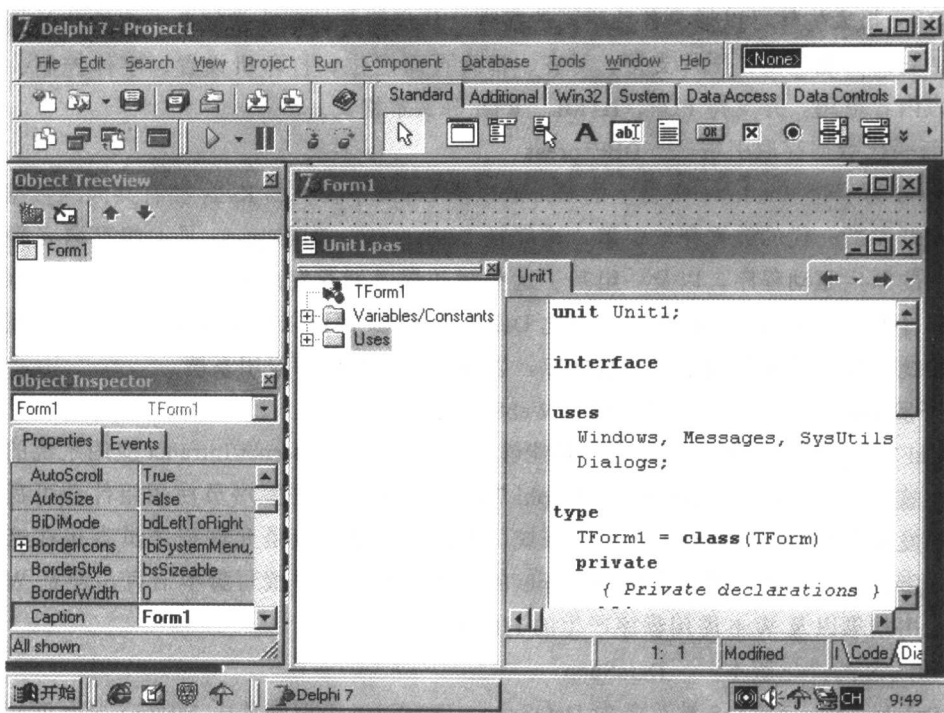


图 1-1 Delphi7 的可视化开发环境

1.2.1 主窗体

主窗体是 Delphi7 可视化开发环境的控制核心，它具有其他 Windows 应用程序的主窗体所具有的一切功能。如图 1-2 所示主窗体包括三部分：主菜单、工具栏和组件面板。

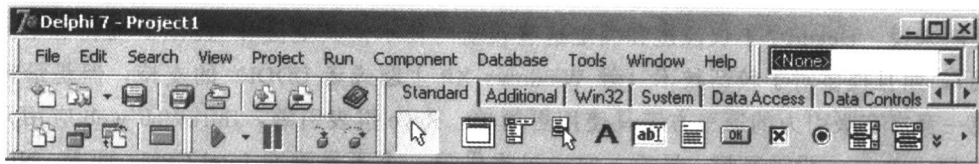


图 1-2 主窗体

1. 主菜单

主菜单提供了进行程序开发所需的命令的功能。与其他 windows 应用程序一样，可以通过主菜单创建、打开或保存文件、调用向导、查看其他窗口、修改选项等等。主菜单的每一项都可以通过工具栏上的一个按钮来实现。使用菜单及子菜单可以完成 Delphi7 中所有的操作功能。表 1-1 列出了菜单的名称和主要功能。

表 1-1 菜单名称和主要功能

菜 单	功 能
File (文件)	用于文件管理，包括新建、打开、保存工程或文件等
Edit (编辑)	用于编辑操作，包括复制、粘贴、撤销等
Search (查询)	用于搜索特定的对象进行替换、定位等

续表 1-1

View (浏览)	用于管理开发环境中的窗口显示
Project (项目)	用于管理项目文件、编译文件及项目设置
Run (运行)	提供调试程序的各种命令
Component (组件)	建立和安装新的自定义组件
Database (数据库)	数据库管理工具, 提供数据库应用程序的各种工具
Tools (工具)	用于设置开发环境选项
Help (帮助)	提供全面的 Delphi7 帮助信息

2. 工具栏

菜单栏中常用的功能被做成图标按钮放在快捷工具栏上, 便于操作, 提高工作效率。工具栏支持自定义功能, 方法是在快捷工具栏上单击鼠标右键, 然后从弹出的快捷菜单中选中或取消所需要的项目, 或选择【Customize】选项进行进一步定制。

IDE 工具栏的定制功能并不仅限于配置需要显示的按钮, 还可以调整工具栏、组件面板和菜单栏在主窗体中的位置。要做到这一点, 只需拖动工具栏右边凸起的灰色条即可。当拖动时, 如果鼠标落在了全窗口区域的外部, 就会看到另一种定制形式: 工具栏可以在主窗体内浮动, 也可以停靠在它们自己的工具窗口内。

3. 组件面板

组件面板是一个带有多个标签页的工具栏, 它包含 Delphi7 中安装的所有组件和 ActiveX 控件。并且按功能分别放置在不同的标签页中。单击页上的标签名, 可以切换页面。可以通过下列方法将组件添加到窗体中:

- 单击所需的组件, 再单击窗体。
- 双击所需的组件, 它就会出现在窗体中。

1.2.2 窗体

窗体是应用程序在设计期间的操作界面。当启动 Delphi7 时, 会在程序开发环境中自动生成空白窗体, 如图 1-3 所示。

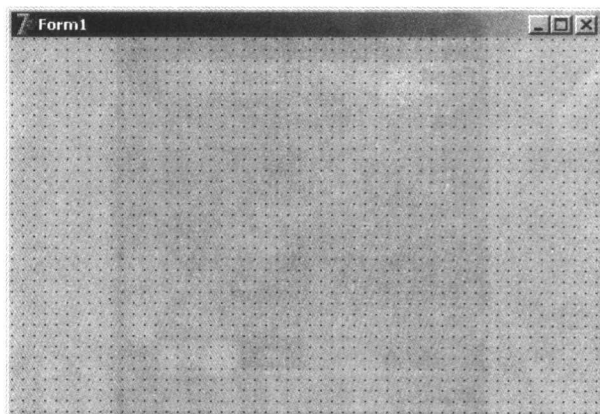


图 1-3 空白窗体

在窗体上可以放置组件，只要从组件板选择一个组件并把它放到窗体上，就可以在窗体上看见该组件，之后可以用鼠标调整组件的位置和大小，还可以用对象监视器来改变组件的属性，用代码编辑器来编写事件处理程序等。

一个窗体对应一个采用 Pascal 语言编写的单元文件（. pas）。通过执行菜单命令 **【View】** | **【Toggle Form】** | **【Unit】** 或快捷键 F12 可以在窗体编辑单元和代码编辑器之间进行切换。

1.2.3 对象树浏览器

对象树浏览器如图 1-4 所示，它以树状图的形式显示了窗体、组件以及数据模块之间的逻辑关系。

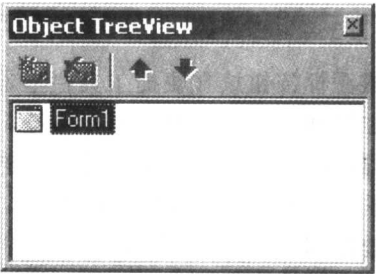


图 1-4 对象树浏览器

1.2.4 对象监视器

对象监视器可以修改窗体或组件的属性，或者使它们能够响应不同的事件。如图 1-5 所示，对象监视器由对象选择器、Properties 页和 Events 页三部分组成。

对象选择器：对象选择器是一个下拉列表框，它显示了当前窗体上所有组件的名称和类型，也包含窗体本身。方便用户在各组件之间进行切换。

Properties 页：属性（Properties）页中列出了当前被选中组件或窗体的所有属性。可以在该页中直接修改某对象属性值。

Events 页：事件（Events）页中列出当前被选中的组件或窗体的所有事件。编写某组件的某个事件处理程序，首先要在窗体上选中某组件，使之成为当前组件，然后在 Events 页上双击该组件要响应的事件，系统就会在代码编辑器中自动生成一个事件处理过程框架，供用户输入源代码。



图 1-5 对象监视器

1.2.5 代码编辑器

代码编辑器是编写程序代码的地方，如图 1-6 所示。Delphi7 的代码编辑器又分为左右两个窗口。左边窗口显示了当前代码编辑窗所有资源的结构列表，包括类定义部分、变量和常量的定义部分以及引用声明部分。右边窗口是真正意义上的代码编辑器，又称为单元窗口，是用来编辑源程序的地方。也是 Delphi7 自动生成代码的地方。代码编辑器也是一个多页结构，每一页对应着一个源代码编辑器或文件。单击每页的标签名，就可以方便地在多个单元间进行切换。

当新建一个窗体后，代码编辑器就会自动生成一个相应的新单元（Unit），里面的内容就是对应于该窗体的源代码文件（.pas）。

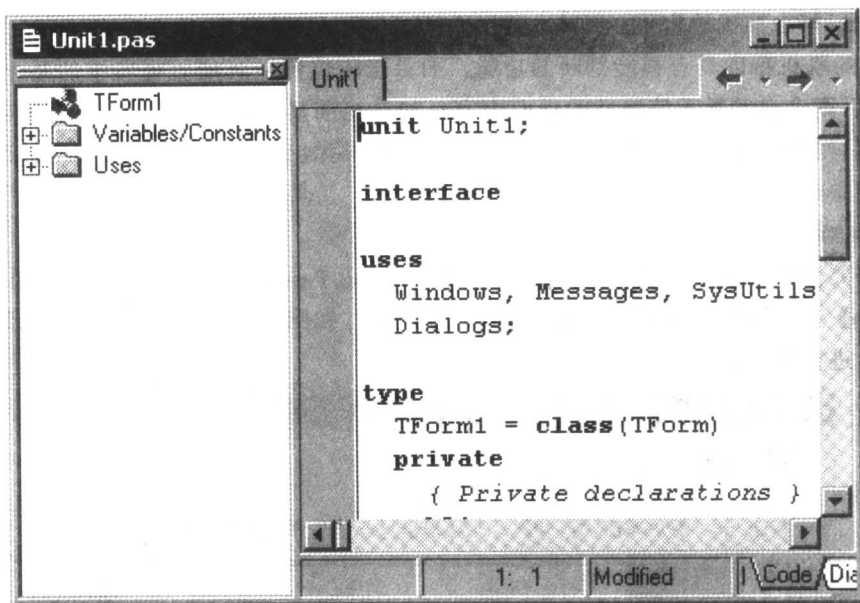


图 1-6 代码编辑器

1.2.6 Delphi 7 的系统功能和帮助

为了方便用户编写应用，Delphi 提供了许多系统功能和帮助。

1. 系统功能

(1) 代码模板

代码模板可以自动提示用户 Delphi 的各种语句的语法。

例如用户输入 if 语句，按 Ctrl+J 键，出现一个弹出式菜单，其中提供有 if 语句类型的选择，用户可以选择把要使用的 if 语句类型，然后按 Enter 键即可输入该类型。

(2) 代码完成功能

代码完成功能将自动提示对象的属性、方法。当用户输入一个后面带有句点的类名或对象名称时，系统会弹出一个菜单，其中包含了与这个对象有关的属性、方法和事件，用户可以从中选择。代码完成功能在极大程度上方便了类属性或方法的输入，并能减少输入错误。

例如当用户输入到 Label1. 并稍停一会，就会出现所示选项；用户选择（用空格键选择了

Caption，则将 Caption 加到 Label1 之后。

(3) 参数提示

参数提示功能可为函数、过程和方法自动提示所需要的参数、参数的顺序及类型，这样就无须浏览代码或联机帮助来找出它的说明，就能知道所需要的参数及参数的顺序。要激活参数提示功能，只需在过程、函数或方法名后键入左括号，稍等几秒钟后就会出现所用参数的类型及参数的顺序，此时可以根据提示输入参数。

2. Delphi 提供了强大的帮助功能。

下面列举了使用帮助系统的一些方法。

(1) F1 键

当遇见问题时，可以尝试按下 F1 键，Delphi 会根据当前的焦点而启动相应的帮助，一般情况下 Delphi 都会准确地定位。

Delphi 的联机帮助具有智能判断的功能。如果当前位置在对象监视器中的某个属性中时，按下 F1 键后则会自动显示该属性的帮助内容；如果当前显示的是 Delphi 系统的某个对话框，按下 F1 键后则会显示该对话框的有关帮助内容；如果弹出了某个菜单，这时按下 F1 键，则会显示该菜单命令的有关帮助内容。

(2) 帮助菜单

如果需要浏览系统的一些帮助内容，比如说 Object Pascal 语言，可以通过菜单【Help】|【Delphi Help】打开【帮助主题：Delphi Help】对话框，如图 1-7 所示，在目录标签中展开相关的条目，进行系统学习。

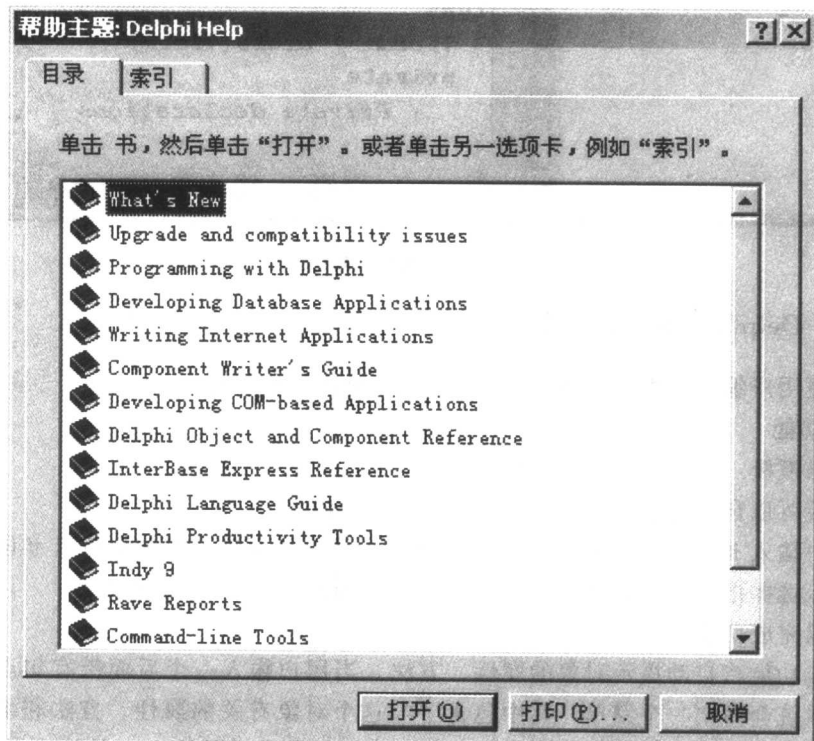


图 1-7 帮助主题

在使用 Delphi 编程的过程中，常常要使用到 Windows API 函数，然而通过菜单命令【Help】|【Delphi Help】打开的【帮助主题：Delphi Help】对话框中，在【索引】标签中可能查不到有关的帮助信息，比如 Windows API 函数 MessageBeep 的有关帮助信息。一种解决的办法就是在代码编辑器中输入 Windows API 函数的名称，然后将光标定位到该名称中，接着按下 F1 键。

另一种办法就是直接打开帮助文件 Win32. hlp，在其中的【索引】标签中查找。可以在 Delphi 的安装光盘上找到帮助文件 Win32. hlp。

在帮助文件打开的情况下，如果要浏览相关内容，可以按下工具栏中的【<<】或【>>】按钮，通常这样是在同一个主题中进行浏览。有些帮助文件中有 Topic group，通过它可以新打开一个主题列表窗口，这样可以非常方便地浏览一个完整主题的帮助内容。此外，通过一些 See else 热链接，也可以查看相关的帮助内容。

在组件栏上单击鼠标右键，通过弹出式菜单的 Help 命令，可以查看有关组件的帮助内容。

(3) 从网上寻求更广泛的帮助

如果在开发过程中遇到不能解决的问题，还可以上网获取有关 Delphi 的信息。我们可以通过下面的网站地址找到相应的 Delphi 的信息。

国外 <http://www.borland.com/delphi/index.html>

国内 <http://www.delphibbs.com> 或 <http://www.csdn.net>

习题 1

1. Delphi 7 的新特性有哪些？
2. Delphi 7 的可视化开发环境由哪几部分组成？它们的功能是什么？