

前 言

Delphi 是 Borland 公司继 Borland C++ 之后又一个使其步入辉煌的开发工具。Delphi 不仅具备超越 Visual Basic 和 PowerBuilder 的可视化快速开发能力,拥有与 Java 和 C# 媲美的面向对象开发机制,而且拥有与 C++ 一样的对 Windows 底层的良好控制能力。

本书吸取国内外教材的先进经验,结合国内高职高专学生的实际情况,坚持做到以丰富的实例来突出 Delphi 的编程特性。教材以语法介绍与可视化设计并重,重点介绍基本理论、基础知识和基本实用技能,强调实用性。为适应高职高专学生的学习,本书在编写过程中注意了难点的分解,本书所选用的实例力求做到典型、生动且易于理解,实例和对应知识点有机结合,图文并茂,代表性强。

本书从案例出发,由浅入深,全面系统地介绍 Delphi 7.0 的基础知识及开发的技巧和應用,展示 Delphi 7.0 强大的功能和易用性。全书共分 11 章。

第 1 章 Delphi 概述:主要介绍 Delphi 7.0 的特点,Delphi 7.0 的集成开发环境、Delphi 7.0 的使用方法等。

第 2 章 Object Pascal 语言:主要介绍 Object Pascal 语言基础,面向对象编程的基本概念和方法等。

第 3 章 常用组件:主要介绍 Delphi 编程中常用的组件和使用方法及案例等。

第 4 章 菜单设计:主要介绍主菜单和弹出式菜单组件,菜单的设计与使用。

第 5 章 窗体设计:主要介绍窗体的主要属性、事件、方法、多窗体技术、利用模板生成各种窗体,多窗体间切换、调用和共享等技术。

第 6 章 图形、图像与多媒体技术:主要介绍 Delphi 7.0 中绘制和处理图形图像所需的基本对象、基本方法及使用技巧。

第 7 章 调试技术与异常处理:主要介绍 Delphi 提供强大的异常处理机制,包括调试器的设置与使用、控制程序的执行、断点设置、监视变量、表达式和属性值等。

第 8 章 数据库编程:主要介绍 BDE 和 ADO 数据集组件中 Table 组件、Query 组件和数据控制组件中的一些组件的属性、事件和方法;并结合实例阐明了 Delphi 中的数据库编程。

第 9 章 报表设计与打印:主要介绍报表设计器 Rave Reports 5.0、Rave 组件、简单报表、多页报表及数据库报表的制作方法。

第 10 章 Internet 编程基础:主要介绍 Internet 的相关概念,重点讲述了 TCP/IP 协议及其常用术语;通过案例讲述了 Web 浏览器的创建与运行,最后通过一个简单的 ISAPI 统计程序,讲述了 Web 服务器的实现。

第 11 章 DLL 的应用:讲述了使用 DLL 的基本知识,介绍了在 Delphi 环境下如何创建和使用 DLL、调用 DLL,以及利用 DLL,实现窗体重用等技术。

本书中所有的程序都是在 Delphi 7.0 环境下调试通过的。

本书适合高职高专学校计算机相关专业教材,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。

本书由张菊担任主编,吴学毅、赵红岩担任副主编,参加编写的有周晓妹、杨艳、单立娟、徐阳、刘继光、张新宇、郝大海。由于作者水平有限,时间仓促,书中难免存在缺点与疏漏,敬请广大读者批评和指正。

编 者
2008 年 1 月

目 录

第 1 章 Delphi 概述	1
1.1 Delphi 简介	1
1.1.1 Delphi 发展史	1
1.1.2 Delphi 的特点	1
1.1.3 Delphi 7.0 的新特点	2
1.2 Delphi 7.0 集成开发环境	3
1.2.1 主窗口	4
1.2.2 对象监视器	6
1.2.3 窗体设计器	7
1.2.4 代码编辑器窗口	8
1.2.5 帮助系统	10
1.3 设计一个 Delphi 7.0 程序	11
1.3.1 新建工程	11
1.3.2 属性设置	11
1.3.3 编辑代码	13
1.3.4 保存工程	14
1.3.5 编译执行	14
1.4 Delphi 7.0 程序的组成	15
1.4.1 程序的基本组成	15
1.4.2 Object Pascal 的单元	18
1.4.3 工程管理	19
本章小结	21
练习与实训	21
第 2 章 Object Pascal 语言	22
2.1 Object Pascal 语言基础	22
2.1.1 基本字符和符号	22
2.1.2 常量与变量	24
2.1.3 运算符与表达式	25
2.1.4 语句	28
2.2 数据类型	30
2.2.1 简单类型	30
2.2.2 字符串类型	33
2.2.3 结构类型	33

2.2.4	指针类型	37
2.2.5	可变类型	38
2.3	三种结构的程序设计	38
2.3.1	顺序控制结构	38
2.3.2	选择控制结构	39
2.3.3	循环控制结构	43
2.4	过程与函数	47
2.4.1	基本概念	47
2.4.2	常用过程与函数	50
2.4.3	子程序的参数	52
2.4.4	子程序的嵌套与递归	55
2.4.5	变量的作用域	56
2.5	Object Pascal 面向 Object 编程基础	57
2.5.1	Object 与类的定义	58
2.5.2	类与单元的关系	58
2.5.3	类的封装与继承	59
	本章小结	60
	练习与实训	61
第 3 章	常用组件	63
3.1	按钮及其分类组件	63
3.1.1	Button 组件	63
3.1.2	BitBtn 组件	64
3.1.3	SpeedButton 组件	65
3.1.4	RadioButton 组件	68
3.1.5	RadioGroup 组件	69
3.1.6	GroupBox 组件	69
3.1.7	CheckBox 组件	72
3.1.8	Panel 组件	73
3.2	文本编辑组件	75
3.2.1	Label 组件	75
3.2.2	Edit 组件	77
3.2.3	MaskEdit 组件	79
3.2.4	Memo 组件	81
3.2.5	RichEdit 组件	82
3.2.6	Listbox 列表框组件	84
3.2.7	组合框组件	87
3.2.8	滑块型组件	87
3.3	工具栏与状态栏组件	90
3.3.1	ToolBar 组件	90

3.3.2	CoolBar 组件	91
3.3.3	StatusBar 组件	91
3.4	显示与输入对话框	92
3.4.1	消息框	92
3.4.2	输入框函数	94
3.5	对话框组件	94
3.5.1	文件对话框组件	95
3.5.2	FontDialog 组件	96
3.5.3	ColorDialog 组件	97
3.5.4	PrintDialog 和 PageSetupDialog 组件	97
3.5.5	FindDialog 和 ReplaceDialog 组件	98
3.5.6	对话框综合应用实例	99
3.6	目录组件	103
3.7	其他组件	105
	本章小结	107
	练习与实训	107
第 4 章	菜单设计	108
4.1	创建主菜单	108
4.2	弹出式菜单	114
4.3	动态设计菜单	114
4.4	使用菜单模板	116
	本章小结	117
	练习与实训	117
第 5 章	窗体设计	118
5.1	窗体属性、方法和事件	118
5.2	窗体设计	120
5.2.1	新建窗体	120
5.2.2	设置窗体属性	121
5.2.3	添加组件	122
5.2.4	设置组件属性	123
5.2.5	保存文件	123
5.3	利用模板生成窗体	123
5.4	多窗体技术	125
5.4.1	MDI 窗体的创建	125
5.4.2	MDI 窗体的属性和方法	126
5.5	窗体间的切换、调用与共享	127
	本章小结	129
	练习与实训	129
第 6 章	图形、图像与多媒体技术	130

6.1	图形图像组件	130
6.1.1	Tcanvas 对象	130
6.1.2	Shape 组件	136
6.1.3	PaintBox 组件	137
6.1.4	时间组件	137
6.2	图像	138
6.2.1	Image 组件	138
6.2.2	Bitmap 组件	138
6.2	动画	139
6.2.1	Animate 功能简介	139
6.2.2	主要属性	139
6.2.3	主要方法	140
6.2.4	主要事件	141
6.2.5	应用举例	141
6.3	多媒体	143
6.3.1	MediaPlayer 组件	144
6.3.2	主要属性	144
6.3.3	常用方法	145
6.3.4	应用举例	146
	本章小结	148
	练习与实训	148
第 7 章	调试技术与异常处理	149
7.1	错误类型	149
7.2	调试器	150
7.2.1	调试前的准备	150
7.2.2	调试器的设置	151
7.3	控制程序的执行	153
7.3.1	运行到指定行	153
7.3.2	单步执行	154
7.3.3	断点执行及断点的设置与使用	154
7.3.4	暂停程序的执行	157
7.4	观察变量、表达式和属性的值	157
7.5	异常处理	158
7.5.1	引入异常处理	158
7.5.2	异常类	158
7.5.3	异常类的触发方法	161
7.5.4	处理异常的方法	162
	本章小结	164
	练习与实训	164

第 8 章 数据库编程	165
8.1 数据库系统	165
8.1.1 基本概念	165
8.1.2 Delphi 中数据库引擎的发展	166
8.1.3 Delphi 数据库开发的步骤	167
8.2 数据库桌面系统	168
8.2.1 Database Desktop 概述	168
8.2.2 设定工作目录与私有目录	169
8.2.3 创建别名	169
8.2.4 创建数据库表	171
8.2.5 数据表格的属性	173
8.2.6 使用数据库窗体向导设计数据库应用程序	175
8.3 BDE 组件	177
8.3.1 BDE 组件常用方法	178
8.3.2 TTable 组件	181
8.3.3 TQuery 组件	186
8.3.4 TStoredProc 组件	191
8.3.5 TDatabase 组件	192
8.3.6 TSession	194
8.4 数据控制组件	197
8.4.1 TDBGrid 组件	198
8.4.2 TDBNavigator 组件	200
8.4.3 TDBText 组件	202
8.4.4 TDBEdit 组件	202
8.4.5 TDBMemo 组件	203
8.4.6 TDBImage 组件	203
8.4.7 TDBListBox 组件	204
8.4.8 TDBComboBox 组件	204
8.4.9 TDBCheckBox 组件	204
8.4.10 TDBRadioGroup 组件	205
8.4.11 其他组件	205
8.5 ADO 编程	205
8.5.1 基本 ADO 的体系结构	205
8.5.2 ADO 数据集组件	206
8.5.3 应用 ADO 开发数据库系统举例	207
8.6 Delphi 中的 SQL 编程	209
8.6.1 SQL 查询语句简介	209
8.6.2 SQL 语言编程中的运用	210
8.6.3 动态 SQL 语句的编程	211

8.6.4	SQL 编程实例	212
8.7	案例：基于 BDE 的学生缴费注册系统	215
8.7.1	数据库设计	216
8.7.2	主控界面设计及属性设置	218
8.7.3	实现自动数据计算功能	221
8.7.4	自己制作数据导航组件	222
8.7.5	制作初始化设置界面	224
8.7.6	增加浏览窗体与任意组合条件的数据统计功能	226
	本章小结	231
	练习与实训	231
第 9 章	报表设计与打印	232
9.1	Rave 报表设计器	232
9.1.1	概述	232
9.1.2	使用 Rave Reports 5.0 创建报表	233
9.1.3	菜单、工具栏和选项卡	236
9.2	Delphi 中的 Rave 组件	240
9.2.1	RvProject 组件	240
9.2.2	RvSystem 组件	241
9.2.3	其他组件	242
9.3	设计简单 Rave 报表	243
9.3.1	案例	243
9.3.2	技术要点	243
9.3.3	设计与实现	243
9.4	设计一个多页报表	245
9.4.1	案例	245
9.4.2	技术要点	246
9.4.3	设计与实现	246
9.5	设计数据库报表	248
9.5.1	案例	248
9.5.2	技术要点	249
9.5.3	设计与实现	249
	本章小结	254
	练习与实训	254
第 10 章	Internet 编程基础	255
10.1	Internet 基本概念	255
10.1.1	网络和 TCP/IP 协议	255
10.1.2	TCP/IP 协议及特点	255
10.2	建立一个 Web 浏览器	256
10.3	创建 Web 服务器	260

10.4 应用举例.....	261
本章小结.....	264
练习与实训.....	264
第 11 章 DLL 的应用.....	265
11.1 DLL 概述.....	265
11.1.1 DLL 的基本概念.....	265
11.1.2 使用 DLL 的优点.....	266
11.2 DLL 的创建.....	266
11.2.1 DLL 工程文件.....	267
11.2.2 DLL 中的 Exports 子句.....	267
11.2.3 编写 DLL.....	268
11.2.4 DLL 的调用.....	270
11.3 DLL 的静态调用方法.....	271
11.4 DLL 的动态调用方法.....	272
11.5 利用 DLL 实现窗体的重用.....	276
11.5.1 创建含有窗体的 DLL 工程文件的步骤.....	276
11.5.2 创建含有窗体的 DLL 工程文件.....	276
11.5.3 调用含有窗体的 DLL 工程文件.....	280
本章小结.....	281
练习与实训.....	281
参考文献.....	282

第 1 章 Delphi 概述

Delphi 是著名的 Borland 公司（现在已和 Inprise 公司合并）开发的一种方便、快捷的 Windows 应用程序开发工具，为程序员提供了全新的可视化编程环境。本章主要介绍 Delphi 的发展历史、Delphi 的特点、面向对象编程的基本知识和 Windows 编程的基本知识。

1.1 Delphi 简介

1.1.1 Delphi 发展史

Delphi 这个名字源于古希腊的城市名，它集中了第三代语言的优点，以 Object Pascal 为基础，扩充了面向对象的能力，并且完美地结合了可视化的开发手段。Delphi 自 1994 年 1.0 推出后就受到了人们的关注，并在当年夺得了多项大奖。

Delphi 的每一代产品都是伴随着 Windows 操作平台的升级而升级，Delphi 1.0 几乎与 Windows 95 同时发行。这对于以前使用 Borland 公司的 Turbo Pascal 语言但又需要开发 Windows 程序的程序员来说提供了极大的方便。Delphi 1.0 可以看作是 Turbo Pascal 的 Windows 版，它为使用 Pascal 语言的用户提供了一个可视化的编程环境。但 Delphi 1.0 仍属于 16 位的开发环境。Delphi 2.0 支持 32 位编程，并且增加了数据模块等多种高效的数据重用功能。Delphi 3.0 改进了集成开发环境，对组件库进行了许多增强，完全支持 ActiveX 控件及其创建，它是一个比较成熟的版本，受到了广泛好评，大大巩固了 Delphi 作为 Windows 应用程序开发工具的地位。Delphi 4.0、Delphi 5.0、Delphi 6.0 不断地改进以前版本的不足，增加新的功能，对开发环境也不断改进和完善，特别是不断增加新组件，以适应网络开发的需要。2002 年 9 月推出的 Delphi 7.0，又在以前的基础上进行了加强，应用非常广泛。

1.1.2 Delphi 的特点

Delphi 采用了当前最先进的编程思想——面向对象的程序设计（Object-Oriented Programming, OOP）。面向对象技术是在编程中逐渐发展起来的，与以前的编程方法相比具有更好的可复用性、可扩展性和可管理性，能更好地适应当今软件开发在规模、复杂性、可靠性、质量和效率上的种种要求，因而越来越多地被推广和使用，是目前公认的主流程序设计方法。

Delphi 提供了大量的可视化组件，可视化编程功能强。一些早期的具有 OOP 性能的程序语言，如 C++、Pascal、SmallTalk 等，虽然具有面向对象的特征，但不能轻松地画出可视化对象，与用户的交互能力较差，程序员仍然要编写大量的代码。Delphi 的推出弥补了这项空

白,程序员不必自己建立对象,只要在提供的程序框架中加入完成功能的代码,其余的都交给 Delphi 去做。Delphi 允许在一个具有真正 OOP 扩展的可视化编程环境中,使用它的 Object Pascal 语言,即可视化编程与面向对象的开发框架紧密地结合起来。对于广大的程序开发人员来讲,使用 Delphi 开发应用软件,会大大地提高编程效率,而且随着应用的深入,将会发现编程不再是枯燥无味的工作。

Delphi 推出了新一代跨平台的数据访问引擎,允许程序使用它连接到各种数据源。Delphi 卓越超群的数据库应用程序开发能力无疑是使其在众多开发工具中脱颖而出的重要“法宝”。到目前为止,应用程序开发工具大体分为两个不同的体系,其中一个体系是传统的程序设计语言(如 Pascal、Basic 和 C 等),用于开发数值运算、系统应用等软件,另一个体系则是通用的数据库应用程序的开发。这两类程序开发工具的发展都极为迅猛,但是二者并没有出现混合渗透迹象,如果使用数据库语言进行传统的算术编程,虽然也能完成相应的功能,但是其编程过程可能极为复杂。如果使用传统的编程语言进行数据库编程,通过调用专用的数据库应用程序接口函数和过程,可能也做得比较完善,但这样往往也极其困难。而 Delphi 恰恰结合了以上两种开发工具的优点,它以传统编程语言 Object Pascal 为基础,完美地结合了数据库语言,既可以用于传统的算术编程又可以用于数据库编程。它的数据处理接口工具 BDE (Borland Database Engine, Borland 数据库引擎)可以用来处理当前流行的数据格式,如 xBase、Paradox 等,也可以通过 BDE 的 SQLLink 直接与 Sybase、SQL Server、Informix、Oracle 等大型数据库连接。可以说,不论是小型数据库应用系统还是关键性、大型数据库应用系统,Delphi 都能满足系统开发的需求,具有强大的扩展能力和数据平台升级迁移的适应能力,能够在 Windows 和 Linux 等平台中使用,而且具有快速的数据处理能力,让程序员能够开发出更有效的数据库应用程序,能适应多种数据库结构。从桌面数据库到客户-服务器模式再到多层数据库结构模式,Delphi 都能胜任。

1.1.3 Delphi 7.0 的新特点

发展至今,Delphi 7.0 在集成开发环境(Integrated Development Environment, IDE)、Web、COM、数据库(Database)、可视化组件(Visual Component Library, VCL)、运行库(Runtime library)、编译器(Compiler)等方面进行了加强。

1. 集成开发环境方面

在集成开发环境方面增加了如下功能:编译信息更全面,可以更好地控制生成的编译信息;组件面板新增了许多功能强大、便于开发网络应用程序的组件;代码编辑进行了改善,按住 Ctrl 键,单击所需查看的方法,就可以方便快捷地查看原始定义;程序员可以在源代码工具中使用不同的颜色,并可以自己创建习惯的代码管理器;调试器可以设置不同的事件日志颜色来显示不同的事件信息,调试过程的记录也更加全面。

2. Web 方面

Delphi 中包含更多的 Web 开发组件,可以方便地开发 Web 服务器应用程序,而且在浏览器、附件类型及其他方面做了增加或者相应的改进。

3. 数据库方面

更新了 dbExpress 的驱动程序,可应用于多种类型及版本,增加了数据库组件。Delphi

的跨平台数据访问引擎 dbExpress 允许程序使用它连接到各种数据源,再配合不同的数据连接 DLL 文件,程序员可以处理后段数据库中的数据。由于 dbExpress 具备了跨平台的能力,能够同时在 Windows 和 Linux 平台以及 .NET 上使用,还提供了快速的数据能力,让程序员能够开发出更有效的数据库应用程序。

4. 可视化组件库及运行库方面

组件库增加了对 Windows XP 的支持。除了支持 Office 97 和 Office 2000 外,许多组件还支持 Office XP。新增了许多组件,部分组件有所更新,加入了一些新的特性。加强了运行库许多单元中函数与过程的功能,有的单元库新增了一些函数与过程。详细内容可以参考 Delphi 7.0 的帮助文件。

12 Delphi7.0 集成开发环境

Delphi 7.0 的集成开发环境 IDE 是用于进行设计、运行和测试程序的工作系统。启动 Windows 后,单击【开始】按钮,指向【所有程序】,选择【Borland Delphi 7】,单击【Delphi 7】运行 Delphi 7.0,运行后可以见到 Delphi 7.0 的集成开发环境,如图 1-1 所示,主要包括以下几个窗口。

主窗口:位于屏幕的顶部,即标题为 Delphi 7-Project1 的窗口。

窗体 (Form) 设计器窗口:位于屏幕的右边,即标题为 Form1 的窗口。

对象监视器窗口:位于屏幕的左下部,即标题为 Object Inspector 的窗口。

对象树形图窗口:位于屏幕的左上部,即标题为 Object TreeView 的窗口。

代码编辑器窗口 (Code Editor):位于窗体窗口的后面,即标题为 Unit1.pas 的窗口。

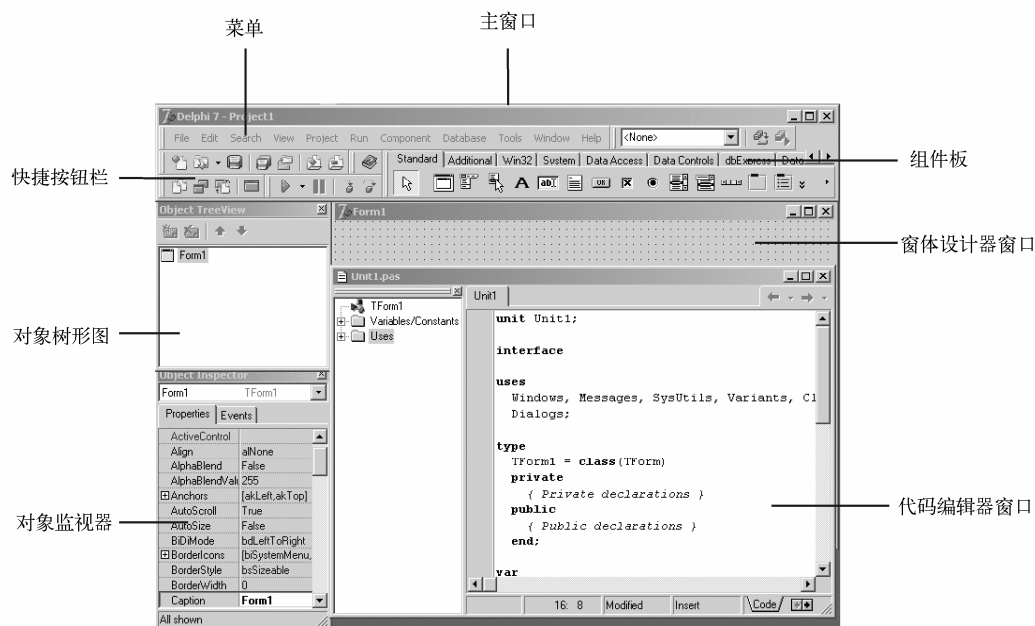


图 1-1 Delphi 7.0 集成开发环境

1.2.1 主窗口

主窗口是 Delphi 可视化开发环境的控制核心,它具有 Windows 应用程序的主窗口应有的一切功能。它包括:标题栏、菜单栏、快捷按钮栏和组件选项板(Component Palette)。Delphi 7.0 的主窗口如图 1-2 所示。

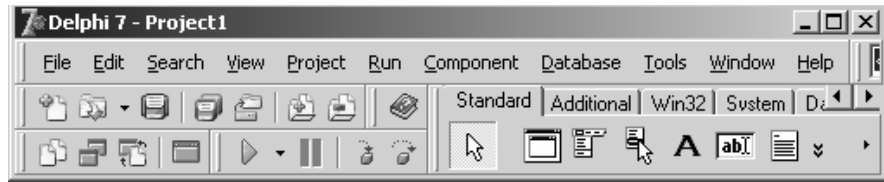


图 1-2 Delphi 7.0 的主窗口

1. 标题栏

主窗口顶部的水平长条称为标题栏。标题栏提供了“ Delphi 7 ”和工程文件名两部分信息。如果载入的是一个新建的工程,Delphi 会提供默认工程名“ Project1 ”,标题栏就会显示“ Delphi 7.0-Project1 ”。

当 Delphi 环境处于不同的状态时,标题栏还会显示相应信息。标题栏右边有 3 个按钮,用于控制窗口的最小化、最大化和关闭。而单击最左侧的小图标,可弹出窗口控制菜单,用鼠标单击就可以完成某些操作。

2. 主菜单

标题栏的下面是菜单栏(Menu),它集合了 Delphi 的各项命令和工具。主菜单可以创建、打开或保存文件、调用向导、查看其他窗口、修改选项,等等。Delphi 主菜单共包括 11 项,如表 1-1 所示。

表 1-1 Delphi 主菜单名称和主要功能

菜单名称	功能
File (文件)	新建、打开和保存项目和文件
Edit (编辑)	编辑代码和窗体组件的各种命令
Search (查询)	搜索、替换和定位字符串命令
View (浏览)	打开 Delphi 环境各个窗口及工程中的窗体和单元文件等
Project (工程)	管理、编译和配置工程文件
Run (运行)	调试应用程序,如设置断点、单步执行等
Component (组件)	用于建立和安装组件,以及定制自己的组件板
Database (数据库)	开发数据库应用程序的各种工具
Tools (工具)	Delphi 环境设置和一些 Delphi 外挂工具
Window (窗口)	切换指定窗口为活动窗口
Help (帮助)	Delphi 帮助文件

3. 快捷按钮栏

快捷按钮栏中以图标按钮的方式显示一些常用的菜单命令，以便于快速访问。每个菜单项对应一个快捷按钮，当鼠标在图标上稍有停顿时会出现功能提示。它向用户提供了比菜单更加快捷直观的操作方式。快捷按钮按其功能分成若干组分别放在相应的工具栏中，如标准栏、系统栏等。工具栏的设置可以由用户自行设置。快捷按钮栏初始状态由 16 个按钮组成，其名称及功能如图 1-3 所示。

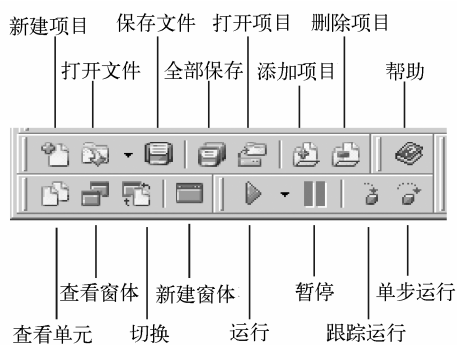


图 1-3 快捷按钮栏

4. 组件板

快捷按钮栏的右侧为组件板。组件板是 Delphi 的核心，它包含丰富的组件，因为 Delphi 主要特点就是使用组件进行程序设计，所以必须掌握各种组件的功能和使用方法。

组件板的组成如图 1-4 所示。



图 1-4 Delphi 7.0 的组件板

组件板包括许多选项卡，单击某些选项卡就可以看到该选项卡包含的组件，图 1-4 中显示了“Standard”选项卡包含的组件。Delphi 7.0 共预定义了三十多个组件选项卡，一般情况下无法全部显示出来。可以单击组件板右边的箭头按钮来左右移动组件板的选项卡标签。

组件板用来选取要添加到窗体上的各种组件，每个选项卡上有一排相关组件的图标，选取要用的图标后，在窗体的某个位置上单击，就可以把它放置到窗体上。表 1-2 列出了常用组件板选项卡的说明。

表 1-2 组件板选项卡

名 称	说 明
Standard	标准的 Windows 控件和菜单
Additional	自定义的控件
Win32	32 位 Windows 的常用组件
System	进行系统访问控制的组件和控件，包括定时器、DDE 等

续表

名称	说明
Data Access	非可视化的组件, 用来访问数据库、数据库表、查询和报表
Data Control	可视化的数据访问控件
ADO	使用 ADO 对象链接数据库的组件
InterBase	链接 InterBase 数据库的控件, 不需要 BDE 和 ADO
Midas	用于多层分布式应用服务的各种控件
Internet Express	用来建立 Web 服务器/多级数据库应用的客户机程序的组件
Internet	用来建立 Web 服务器应用程序的对象
Decision Cube	总结数据库中信息的控件, 用来分析数据库中的数据
Dialogs	Windows 的常用对话框
Win3.1	与 Delphi 1.0 项目兼容的组件
Samples	自制的组件例子, 包括进度指示器、颜色网格等
ActiveX	ActiveX 控件的例子
Servers	封装 COM 服务器

1.2.2 对象监视器

窗体窗口的左侧为对象监视器, 它的功能是编辑设置窗体、组件的属性, 并为窗体、组件添加事件的响应处理过程, 如图 1-5 所示。

1. 对象监视器窗口

对象监视器由标题栏、对象选择器、属性选项卡和事件选项卡组成。

- ☞ 标题栏: 显示“Object Inspector”字样。
- ☞ 对象选择器: 为一个下拉列表框, 列表框中是可供选择的窗体及当前窗体上放置的所有组件。用户可从中选择要设置属性或要添加事件的窗体或组件。
- ☞ 【Properties】(属性)选项卡: 用于设置和查看当前组件的属性, 随着对象选择器中的所选组件而发生变化。属性选项卡是一个两列的表格, 左列是属性标题, 右列是属性值。只有选择了属性标题, 才可以设置属性值。
- ☞ 【Events】(事件)选项卡: 用于为当前组件添加各种响应事件(如单击、拖曳事件等), 也随对象选择器中的所选组件而发生变化。事件选项卡同样也是一个两列的表格, 左列是事件名称, 右列是事件处理过程, 默认时的事件处理过程为空, 在空格处双击则可以为当前组件添加事件, 并进入代码编辑器窗口。

2. 设置属性

从窗体上选择组件, 或从对象选择列表框中选择窗体或组件, 使之成为当前组件, 就可以用对象监视器的属性选项卡来设置其属性了。根据属性标题的情况, 设置属性可分为 4 种情况。

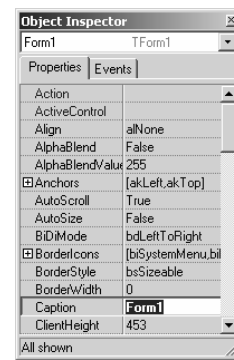


图 1-5 对象监视器窗口

第 1 种情况是用鼠标单击属性标题后,直接在属性值编辑框中输入属性的数值或字符串,如 Name、Caption、Left、Top 等属性,如图 1-6 (a) 所示。

注意:如果输入的属性值不符合要求,例如,在应输入数值的属性中输入了字符串,Delphi 将给出报错信息。

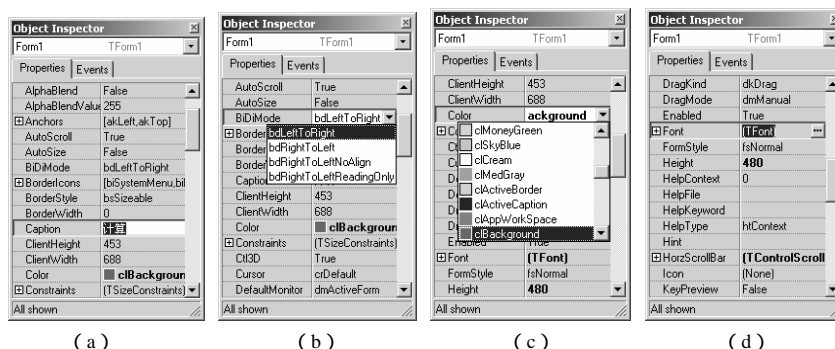


图 1-6 对象监视器属性选项卡

第 2 种情况是属性值由 Delphi 提供可选项。这时,用鼠标单击属性标题后,属性值马上变成一个下拉列表框,用户可以从中选择所需要的属性值。可选项分为两种,一种是文字可选项,如 Border Style、Enabled 等属性;另一种是图形加文字形式的可选项,如 Color、Cursor 等属性。图 1-6 (b) 和图 1-6 (c) 显示了这两种可选项的属性值形式。

第 3 种情况是通过对话框来设置属性值。对于 Font、Color 等属性,单击属性标题后,属性右端就会出现一个带“...”的按钮,单击该按钮,即可打开一个相应的对话框,在对话框中选择所需的值,即可设置属性值,如图 1-6 (d) 所示。

第 4 种情况是属性带有子属性。在这种属性的属性标题前,一般会有一个“+”号,单击该“+”号,将展开子属性,同时“+”号变成“-”号。此时即可依次对各子属性进行设置,如图 1-6 (d) 所示。

3. 添加事件

从窗体上选择一个组件,或从对象监视器中选择一个窗体或组件,使之成为当前组件,就可以用对象监视器的事件选项卡来添加事件了。

添加事件一般是选定组件后,单击对象监视器的事件选项卡中要添加的事件名称,再在其右侧输入事件的相应函数名称,然后按 Enter 键即可。

提示:如果单击要添加的事件后,在其右侧编辑框中双击鼠标,将为事件采用默认事件处理过程名,同时进入代码编辑窗口。

默认事件处理过程名是组件名和事件名去除“On”后合并而成的字符串,如 Button1 组件的 OnClick 事件的默认事件处理过程名是 Button1Click。

1.2.3 窗体设计器

窗体设计器窗口也称窗体窗口,是用于设计应用程序的界面,在大多数情况下,窗体是

用户的主要操作对象。窗体设计器的外观如图 1-7 所示。

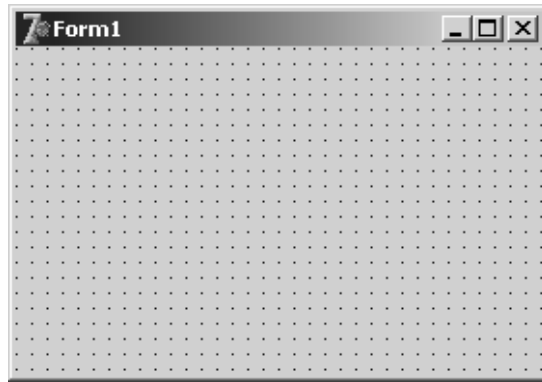


图 1-7 窗体设计器

1. 窗体窗口

窗体设计器的顶端水平条是窗体的标题栏，结构与主窗口的标题栏相似，由图标、窗体名称和用于控制窗体的最小化、最大化和关闭按钮组成。

窗体设计器的标题栏显示了窗体的当前标题。一般新窗体会用默认名字 Form1 命名。当然，用户可以通过对象监视器设置窗体的标题等属性。

窗体是窗口的设计界面，用户可以向窗体重添加各种组件，然后再对组件的属性进行设置。这样可使整个程序的编辑过程可视化。

2. 窗体文件

每一个窗体都有相应的窗体文件（.DRM），它保存着窗体和窗体内各个组件的属性，而且每个窗体都有一个单元文件，用于编写窗体事件处理过程。窗体文件和单元文件被 Delphi 自动保存在同一个目录下。

用鼠标右键单击窗体网格工作区，会打开一个快捷菜单。该菜单包含了设计窗体时所需的各种常用命令。

选择“View as Text”命令可以看到“.DFM”为扩展名的窗体文件。

窗体文件列出了窗体的类名（TForm1）及窗体各属性的值，同时还列出了窗体中包含的组件及其属性值。用户每修改一次窗体，窗体文件都会有所反应。当然，也可直接通过修改窗体文件来设置属性，只不过因为太复杂，很少采用。

在窗体文件空白处单击鼠标右键，将打开一个快捷菜单，从中选择“View as Form”命令即可恢复窗体的图形化显示。

1.2.4 代码编辑器窗口

窗体设计完成后，就要开始编辑代码来响应窗体及其组件的事件了。代码编辑器是输入代码编写程序的地方，也是 Delphi 自动生成代码的地方。代码编辑器由左边的代码浏览器和右边的多页卡代码编辑器窗口组成。代码编辑器的外观如图 1-8 所示。