

跟我学电脑软件系列丛书

跟我学 Delphi 5.0

编程指南



卓越文化艺术有限公司 策划
朱宏伟 主编

初学者的实用宝典
程序员的必备参考

Borland
Delphi 5

冶金工业出版社

跟我学 Delphi 5.0 编程指南

主 编 朱宏伟

编 委 刘 春 张 蕾

邱玉力 王立良

北 京

冶金工业出版社

2000

内 容 提 要

本书是一部 Delphi 5.0 的最新力作, 通过具体的实例说明 Delphi 5.0 的基本概念和使用 Delphi 编程的技巧。

全书共包括 12 章: 第 1 章是 Delphi 5.0 的简介; 第 2~5 章介绍了窗体、工具栏和状态栏、画布、Registry; 第 6 章阐述了 Delphi 的消息处理机制; 第 7 章讲述了如何编写 DLL (动态链接库); 第 8 章讲述了多媒体应用; 第 9 章作为第 8 章的延续, 讲述了如何实现 MP3 播放器; 第 10 章讲述了数据库; 第 11 章讲述了网络应用程序; 第 12 章讲述了文件的操作。本书不仅阐述了 Delphi 编程的主要方面, 同时也针对某些细节作了详细的实例说明。

本书叙述简明易懂, 适用于广大计算机爱好者和学习使用 Delphi 的读者。

图书在版编目 (CIP) 数据

跟我学 Delphi 5.0 编程指南/ 朱宏伟编著. —北京: 冶金
工业出版社, 2000.1
ISBN 7-5024-2515-2

I. 跟… II. 朱… III. DELPHI 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (1999) 第 72341 号

跟我学 Delphi 5.0 编程指南

出 版 人 卿启云 (北京沙滩嵩祝院北巷 39 号)
主 编 朱宏伟
责任编辑 肖 放
封面设计 卓越公司
版式设计 崔亚海

出 版 冶金工业出版社 (邮编 100009)
发 行 冶金工业出版社发行部
经 销 全国新华书店
印 刷 北京云浩印刷厂
开 本 787×1092 16 开 19.25 印张 458 千字
版 次 2000 年 1 月 第 1 版
2000 年 5 月 第 2 次印刷
印 数 8001-12000
书 号 ISBN 7-5024-2515-2/ TP·138
定 价 26.80 元

冶金工业出版社发行部电话: (010) 65934239 64044283
邮购部电话: (010) 65934239 传真 (010) 64013877

前 言

对于一位编程人员来说，好的编程开发软件无疑是最大的好帮手，Delphi 5.0 就是其中之一。

Delphi 5.0 是由著名的 Inprise 公司（即前 Borland 公司）开发的可视化软件开发工具软件。它基于 32 位的 Windows 操作系统，具有高效、可视化的特点。它方便的可视化应用程序开发环境和先进的数据库技术受到了编程人员和计算机爱好者的好评。

Delphi 提供了多种开发工具。包括集成开发环境、Database Desktop、Database Expert 等应用程序。Delphi 的最大优点在于它拥有一个封装完备的可视组件库——VCL。

Delphi 使用的是面向对象的 Pascal 语言（Object Pascal）。Delphi 不仅编译速度非常快，还优化了对数据库的操作，提供了强有力的数据库引擎。

与最初的 Delphi 1.0 相比，今天的 Delphi 5.0 已经发展得非常成熟。它在应用程序界面、数据库、网络等方面都有了很大提高。

本书的最大特点是突出重点，通过具体的实例说明基本概念和使用 Delphi 编程的技巧。由于 Delphi 5.0 覆盖内容非常广，因此，很难全面详细地介绍其中的每一个内容。我们希望通过一些技巧和实例使读者掌握 Delphi 编程的精髓。如果你能对本书的实例和技巧仔细琢磨，相信你会获益匪浅。

如果你是一名具有中级或高级水平的 Delphi 程序员，那么本书的实例和技巧将会让你开阔眼界；如果你是一位初学者，也不必担心，通过本书循序渐进的实例和技巧，你将对 Delphi 编程产生浓厚的兴趣，这一点非常重要，因为兴趣是通往成功的桥梁。

由于本书以实例贯穿始终，初学者甚至一些具有中级水平的读者容易急于求成，不认真阅读和理解实例和技巧的精髓，而是一味地将代码录入到电脑中，看到奇妙的结果就欣欣然了。希望读者在理解程序的基础上，进行上机尝试，这样一定会收到很好的效果。

本书由北京卓越文化艺术有限公司总策划，朱宏伟主编。由于作者水平所限，书中难免会出现一些错误，希望读者谅解并指正。

编 者

1999 年 12 月

目 录

第 1 章 Delphi 5.0 简介	1
1.1 Delphi 5.0 的集成开发环境	1
1.1.1 主窗口	1
1.1.2 对象管理器	3
1.1.3 窗体	5
1.1.4 代码编辑器	6
1.2 如何建立 Delphi 程序	7
1.2.1 创建窗体	8
1.2.2 添加组件	8
1.2.3 设置属性	8
1.2.4 响应事件	8
1.2.5 调试运行	9
1.3 Delphi 文件分析	10
1.3.1 项目文件	11
1.3.2 单元文件	11
1.3.3 窗体说明文件	12
1.4 Delphi 5.0 新增特性	14
第 2 章 了解窗体	15
2.1 如何改变窗体的属性	15
2.1.1 使用对象管理器	15
2.1.2 通过窗体的 OnCreate 事件改变窗体属性	16
2.2 如何创建新的窗体	18
2.2.1 静态创建新的窗体	18
2.2.2 动态创建新的窗体	19
2.3 如何结束应用程序	21
2.3.1 关闭主窗体	21
2.3.2 使用 TApplication 的 Terminate 方法	22
2.3.3 使用 Halt 方法	23
2.4 结束应用程序前如何提示用户	25
2.4.1 在 OnClose 事件中使用对话框	25
2.4.2 在 OnCloseQuery 事件中使用对话框	26
2.5 使程序启动时不显示窗体	27

2.5.1	修改窗体的属性.....	28
2.5.2	调用 ShowWindow 函数.....	28
2.5.3	将窗体隐藏在任务条右下角的标题栏中.....	29
2.6	如何限制窗体最大化时的大小.....	33
2.6.1	重写窗体的 OnResize 事件.....	33
2.6.2	捕获 WM_GETMINMAXINFO 消息.....	35
2.7	如何划分窗体.....	36
2.7.1	利用 Splitter 控件划分窗体.....	37
2.7.2	利用鼠标事件划分窗体.....	38
2.7.3	利用 HeaderControl 控件划分窗体.....	39
2.7.4	利用 Header 控件划分窗体.....	42
2.7.5	利用 Label 控件划分窗体.....	43
2.8	如何使窗体的标题栏闪动.....	44
2.8.1	改变当前窗体.....	44
2.8.2	调用 API 函数 FlashWindow.....	46
2.9	如何拖动窗体的任意区域移动窗体.....	47
2.9.1	直接捕获鼠标事件.....	47
2.9.2	捕获 WM_NCHITTEST 消息.....	49
2.10	如何显示多页窗体.....	49
2.10.1	利用 Notebook 和 TabSet 控件.....	49
2.10.2	利用 PageControl 控件.....	51
2.11	防止应用程序的同时多次运行.....	52
2.11.1	利用 API 函数 FindWindow.....	52
2.11.2	利用互斥对象.....	53

第 3 章 工具栏及状态栏.....57

3.1	ToolBar 组件.....	57
3.1.1	使用 ToolBar 组件制作工具栏.....	57
3.1.2	ToolBar 运用实例.....	60
3.2	CoolBar 组件.....	67
3.2.1	添加 CoolBar 组件.....	68
3.2.2	向 CoolBar 添加其他组件.....	68
3.2.3	CoolBar 的特点.....	69
3.3	StatusBar 组件.....	69
3.3.1	添加 StatusBar 组件.....	69
3.3.2	在状态栏中添加提示信息.....	70
3.3.3	在状态栏中显示按键信息.....	70

第 4 章 画布对象 Canvas	72
4.1 画简单的图形	72
4.1.1 传统的 Windows 绘图方法	72
4.1.2 利用 Canvas 对象绘图	73
4.2 如何显示位图	74
4.2.1 利用 Image 控件	74
4.2.2 利用 Bitmap 对象显示位图	75
4.3 分块显示位图	76
4.3.1 利用 Canvas 对象的 CopyRect 方法	77
4.3.2 利用 Canvas 对象的 Brush 属性	78
4.4 改变位图中特定的颜色	79
4.4.1 利用 Canvas 对象的 Pixels 数组属性	79
4.4.2 利用 Canvas 对象的 BrushCopy 方法	80
4.5 透明地显示位图	81
4.5.1 利用 Bitmap 对象的 Transparent 属性	81
4.5.2 利用位图的逻辑运算产生透明的位图	83
4.6 在位图上显示文字	86
4.6.1 利用 Canvas 对象的 Pixels 属性	86
4.6.2 改变 Brush 对象的 Style 属性	88
4.7 颠倒显示位图	89
4.7.1 利用 Canvas 对象的 Pixels 属性逐行颠倒	89
4.7.2 巧妙利用 Rect 的参数顺序	90
4.8 位图显示特技集锦	92
4.8.1 使位图从下往上逐渐显示在窗体上	92
4.8.2 使位图由小变大逐渐显示在窗体上	94
4.8.3 使位图以马赛克效果显示在窗体上	95
4.8.4 使位图以淡入淡出效果显示在窗体上	98
4.8.5 两幅位图融合的效果	101
第 5 章 注册表操纵	104
5.1 TRegistry 组件	105
5.1.1 TRegistry 组件的属性	105
5.1.2 TRegistry 组件的方法	106
5.1.3 清除 Windows 95 中添加/删除程序遗留的“垃圾”	108
5.2 TRegIniFile 组件	110
5.2.1 TRegIniFile 组件的特殊功能	111
5.2.2 TRegIniFile 组件的属性和方法	111
5.2.3 应用实例	112

第 6 章 消息处理	121
6.1 Delphi 消息系统	121
6.2 覆盖原有消息处理方法	121
6.2.1 简要操作步骤	122
6.2.2 范例	122
6.3 定义用户消息及声明消息处理方法	124
6.3.1 定义消息	125
6.3.2 声明新的消息处理方法	126
6.3.3 发送自定义消息	126
6.3.4 范例	126
第 7 章 学习使用 DLL	130
7.1 通过 DLL 实现字符串的显示	130
7.1.1 静态装载 DLL	130
7.1.2 动态装载 DLL 程序	133
7.2 在 DLL 中使用窗体	134
7.2.1 使用普通的 Form	135
7.2.2 在 DLL 中使用 MDI CHILD Form	137
7.3 在 DLL 中使用资源	140
7.3.1 在普通的应用程序中使用资源	141
7.3.2 在 DLL 中使用资源	142
第 8 章 多媒体应用	146
8.1 AVI 文件的播放	146
8.1.1 实现步骤	147
8.1.2 程序源代码及分析	150
8.2 MediaPlayer 组件	151
8.2.1 MediaPlayer 组件的主要属性	151
8.2.2 MediaPlayer 组件的响应事件	157
8.2.3 MediaPlayer 组件的重要方法	159
8.3 制作一个多媒体播放器	162
8.3.1 实现多媒体播放器	162
8.3.2 源程序清单及分析	166
8.4 Animate 组件及动画运用	172
8.4.1 动画的播放	172
8.4.2 Animate 组件	174

第 9 章 MP3 播放器	177
9.1 利用现有软件内核代码的技巧	177
9.1.1 获取 WINAMP 的窗口句柄	177
9.1.2 确定 Winamp 执行各项功能所对应的消息	179
9.1.3 编写消息的发送方法	179
9.2 创建组件	189
9.2.1 建立单元文件和选择基类	190
9.2.2 为组件类添加属性、方法、事件	190
9.2.3 注册组件	194
9.2.4 源代码清单	195
9.3 利用组件编写简单的播放器	204
第 10 章 Delphi 数据库应用程序开发	208
10.1 Delphi 数据库系统	208
10.1.1 数据库管理系统 (Database Management System)	208
10.1.2 数据库应用程序	209
10.1.3 Delphi 5.0——强大的数据库应用程序开发工具	209
10.2 关于开发 Delphi 5.0 数据库应用程序	209
10.3 设计基于 TTable 组件的同学录	212
10.3.1 实例简介	212
10.3.2 相关组件简介	212
10.3.3 具体设计过程	214
10.4 基于 TQuery 组件的 SQL 应用程序	222
10.4.1 实例说明	222
10.4.2 相关组件介绍	222
10.4.3 SQL 语言基础	223
10.4.4 实例设计	226
10.5 基于 Client/Server 的数据库备份	233
10.5.1 实例说明	233
10.5.2 组件介绍	233
10.5.3 编程思路	238
10.5.4 界面设计	239
10.5.5 源程序及注释	239
第 11 章 网络应用程序	249
11.1 聊天程序	249
11.1.1 使用 TCP 方式	250
11.1.2 使用 UDP 方式	255

11.2	Web 应用程序	258
11.2.1	创建静态网页	259
11.2.2	创建动态 Web 页面	261
第 12 章	文件管理	268
12.1	文件类型和标准方法	268
12.1.1	文本文件	269
12.1.2	有类型文件	274
12.1.3	无类型文件	275
12.2	与文件有关的 Delphi 组件	276
12.2.1	Win 3.1 选项页中的文件支持组件	277
12.2.2	Dialogs 选项页中的文件支持组件	279
12.2.3	Sample 选项页中的文件支持组件	281
12.3	文件管理器的设计	281
12.3.1	实现文件管理器的步骤	281
12.3.2	程序源代码	285

第1章 Delphi 5.0 简介

在本章中，我们将简要地介绍 Delphi 5.0 的集成开发环境，让读者了解一个 Delphi 程序的开发过程以及 Delphi 的项目文件的大致内容，最后说明 Delphi 5.0 的新增特性。通过本章的介绍，初学者将对 Delphi 5.0 有一个总体的认识，对继续深入学习作好准备。

1.1 Delphi 5.0 的集成开发环境

运行 Delphi 5.0，进入了 Delphi 5.0 的集成开发环境，其主界面如图 1-1 所示。

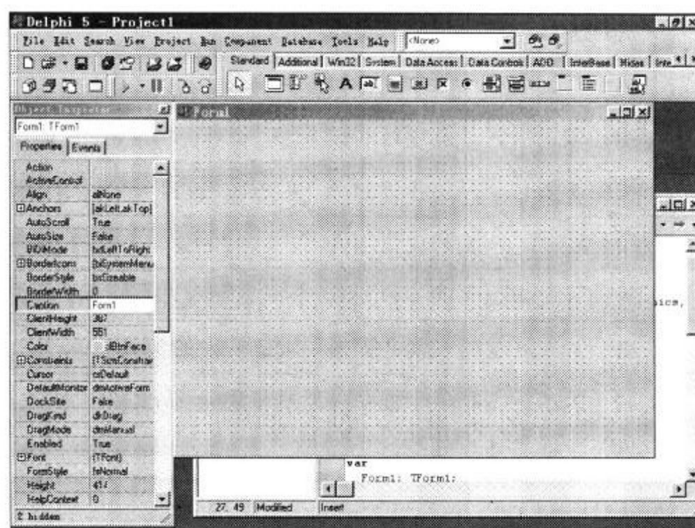


图 1-1 Delphi 5.0 的集成开发环境

Delphi 5.0 的开发界面包括了 4 个部分：主窗口、对象管理器、窗体和代码编辑器。

1.1.1 主窗口

主窗口由标题栏、菜单栏、工具栏和组件板 4 部分组成，如图 1-2 所示。

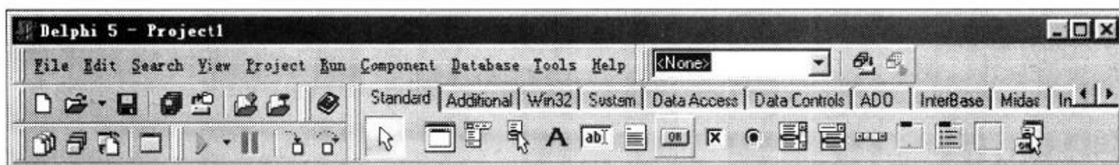


图 1-2 主窗口

主窗口管理着整个开发环境的操作和设置，因此，熟练地掌握和使用它将大大提高开发程序的效率。

1. 标题栏

标题栏将显示当前的项目名称。项目（project）及其包含的文件将在后面介绍。

2. 菜单栏

各菜单项中包含了 Delphi 5.0 开发环境的所有设置和命令，它包含的内容很多，这里就不一一说明。读者可在以后的学习和实践中逐步掌握。下面给出各菜单项的名称和主要功能供读者参考，见表 1-1。

表 1-1 菜单简介

菜单名称	功 能
File	提供项目和各个单元的文件操作命令，如新建、打开、保存等
Edit	提供编辑代码和组件的命令，如复制、粘贴、删除等
Search	提供查找、替换代码等命令
View	用于打开集成环境中的各种工具窗口
Project	用于项目的管理及项目的设置等
Run	提供了调试程序的各种命令，如设置断点、单步执行等
Component	用于建立、安装和设置组件
Database	提供开发数据库的各种工具
Tools	用于开发环境的设置并提供了辅助开发的工具
Workgroups	用于版本控制和多个程序员项目管理
Help	提供了全面的帮助信息

3. 工具栏

工具栏为各种操作提供了快捷按钮，方便用户的使用，如图 1-3 所示。

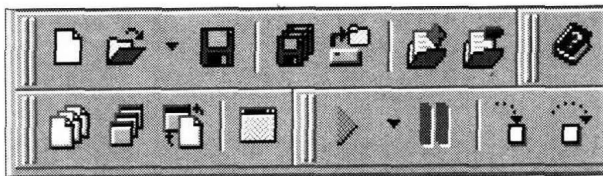


图 1-3 工具栏

读者可以通过工具栏的提示功能来了解各按钮的功能。

4. 组件板

组件板是 Delphi 可视化编程中的核心部件，它包含了许多的选项页，如 Standard、Additional、Win32、System 等。每个选项页中又包含了若干组件。使用者通过选取组件，并将组件添加到窗口来进行程序的初步设计。

组件板如图 1-4 所示。

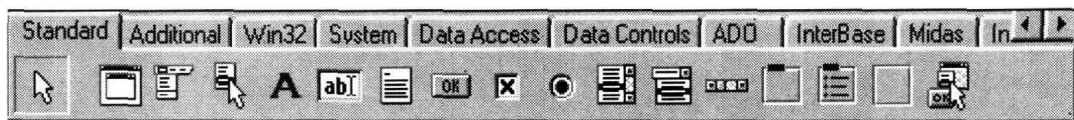


图 1-4 组件板

1.1.2 对象管理器

对象管理器用于组件对象的属性设置和建立事件响应。当往窗体中添加完组件对象后，就可以通过对象管理器来进行程序的开发。

对象管理器的顶端是一个下拉的对象列表框，可以从中选择所需的对象进行设计。当窗体中某一组件处于焦点时（即被选中），对象管理器将自动显示该对象的属性和事件。对象管理器中有两个选项页：Properties（属性）页和 Events（事件）页。这两个选项页分别用于设置组件对象的属性和建立组件对象的事件响应。

对象管理器如图 1-5 所示。

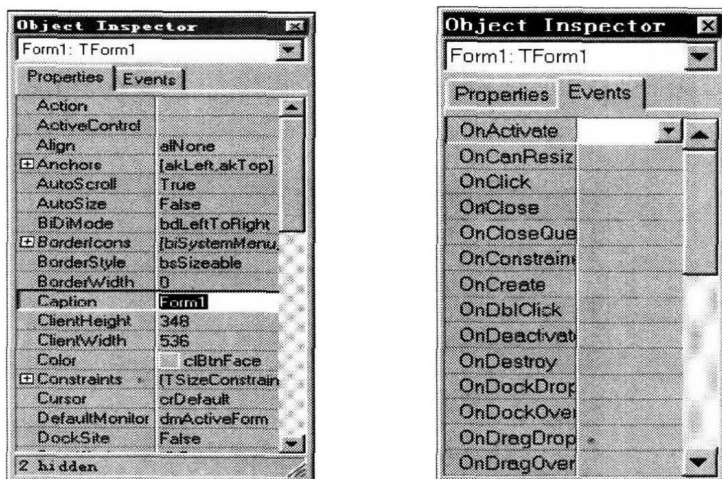
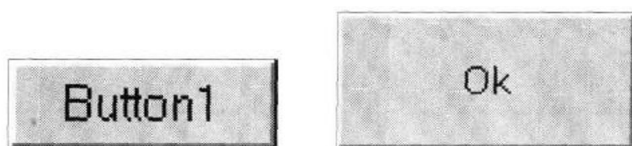


图 1-5 对象管理器（Properties 和 Events 选项页）

1. 属性

每个组件都有一系列的属性，如对象的名称（Name）、文字标题（Caption）、高度（Height）、宽度（Width）等。管理器的属性栏中列出了一个组件对象的大部分但并非全部的属性，每项属性的左边格子为属性名，右边格子中是当前的属性值，使用者可以根据自己的需要来设置这些属性值。另外还有一些属性是不能在程序设计时进行设置的（只在程序运行时设置），或是只用来存储状态数据，根本不能被使用者设置（即只读属性）。因此这两种属性都不出现在管理器的属性栏中。一般来讲，属性决定了组件对象的外观，下面举例说明。

例如，在窗体中添加了一个 TButton 对象 Button1，如图 1-6（a）所示。我们将 Button1 的 Caption 属性改为“OK”，并用鼠标拖动 Button1 的边界线来改变其大小，此时 Height 属性和 Width 属性将自动被更改。当然，你也可以直接地设置 Height 和 Width 属性来改变按钮的大小。改变后的结果如图 1-6（b）所示。



(a)

(b)

图 1-6 Button1 的属性设置

2. 事件

事件选项页中列出了组件对象可以触发的所有事件。对于每个事件，左边是事件名称，右边是响应该事件的过程名，当未设置事件响应时，右边是空白框。

什么是事件呢？事件是一个句柄，是一个过程类型变量，更通俗地讲，事件是一个指向过程的指针。所谓的设置事件响应就是把一个具体的过程名赋给事件，就是一个过程类型的赋值操作，或者说是一个指向过程的指针的赋值操作，即事件决定了组件对象。

举例说明，通常在程序设计中需要响应鼠标单击按钮的操作，这样就必须设置按钮对象的 OnClick 事件。按钮对象为 Button1，我们双击 OnClick 右边的空白框，Delphi 就自动在单元文件中添加一个过程“Button1Click”，并把该过程的过程名“Button1Click”赋给了 OnClick 事件，结果如图 1-7 所示。

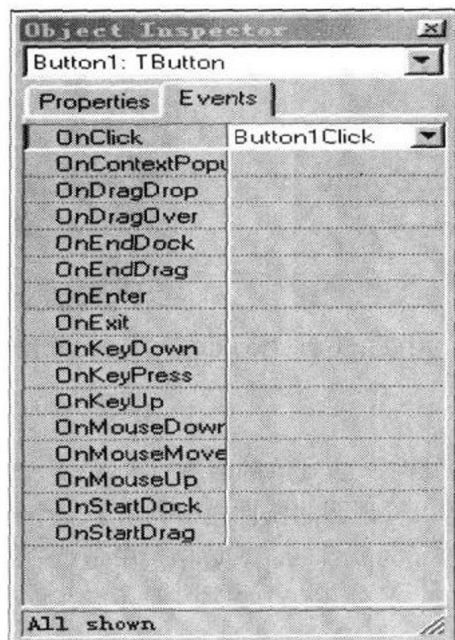


图 1-7 设置 OnClick 事件

这样 OnClick 事件的设置就完成了。当然，还需要在过程中添加代码，否则只是个空过程，如图 1-8 所示，不进行任何处理。事件响应的全过程是这样的：当单击按钮时，OnClick 事件被触发，由于此时的 OnClick 已被设置指向过程 Button1Click，将调用 Button1Click 过程，执行其中的代码。

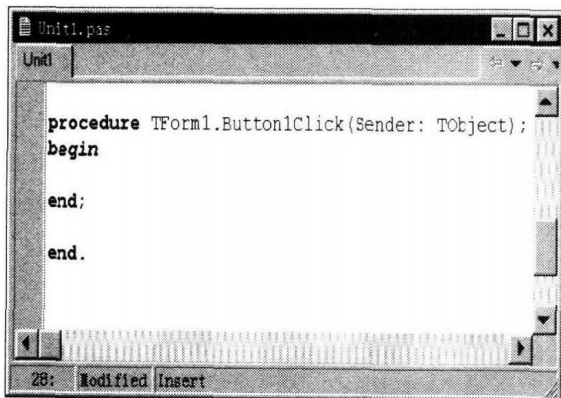


图 1-8 设置 OnClick 事件

注意：从本质上讲，事件也是一种属性。这一点可以从 Delphi 的帮助中看出，对属性和事件的定义都使用了“property”这个关键字。

如 OnClick 事件的声明如下：

Property OnClick: TNotifyEvent;

前面已说过，事件是一个过程类型变量，TNotifyEvent 就是一个过程类型。因此，在本质上它与属性并无区别，它们都是类（Class）中定义的数据。将二者区分开来，主要是出于使用和功能上的考虑。

1.1.3 窗体

启动 Delphi 后，Delphi 会自动产生一个空白的新窗体，如图 1-9 所示。运行时，窗体是一个程序的界面；在设计时，窗体是一个组件容器，可以在上面任意添加组件。对于可视化编程来说，程序设计总是由窗体开始的。

窗体（Form）和窗口（Window）有何区别？可以肯定地说，窗体就是一个窗口。这点毫无疑问，因为作为一个程序的主界面，窗体必然是一个窗口。但是，窗口不一定是窗体。这里的窗口是指广义的窗口，多数的 Delphi 组件都是窗口，如按钮，列表框等都是窗口。窗体则是指程序的主窗口。

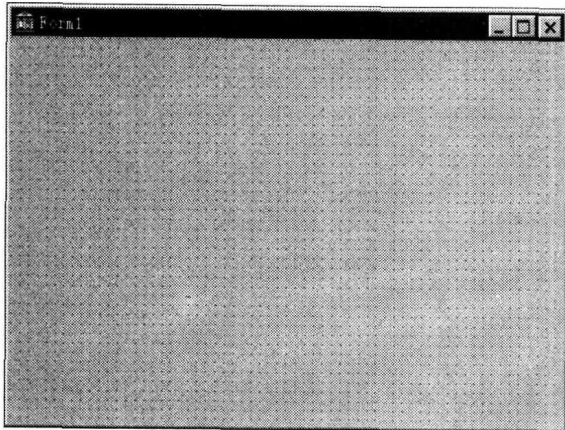


图 1-9 窗体

1.1.4 代码编辑器

代码编辑器用于程序代码的编辑，它可以打开项目中的所有单元文件，使用者可以使用它来添加自己的程序代码。可以说，代码的编写是整个程序设计的核心。尽管可视化编程技术的运用大大地减轻了程序员编写代码的工作量，但它并不能取代原始的代码编写工作。一个程序的好坏将大部分取决于程序代码的编写。

Delphi 启动后，在自动生成一个新窗体的同时，还生成了一个与之相应的单元文件并显示在代码编辑器中，如图 1-10 所示。

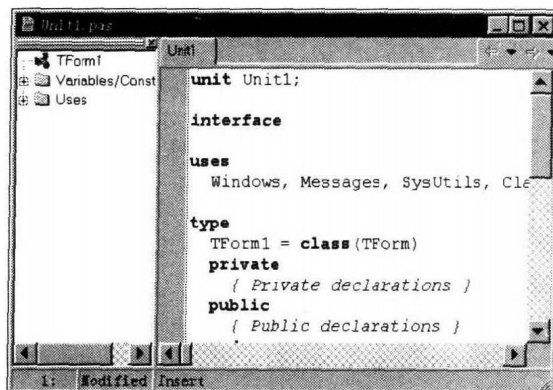


图 1-10 代码编辑器

Delphi 的代码编辑器使用起来非常方便，它能提示和帮助用户完成代码的编写。这就是 Delphi 中的 Code Insight 技术。用户不用担心记不住大量的属性、事件、方法以及方法中的参数，Delphi 会随时提示你。

1. 代码完成向导

当用户在编辑器中键入了一个后面带有句点的类或类的对象后，稍作停顿，Delphi 会自动弹出一个提示框，其中包含了该类的属性、事件和方法。例如，用户键入“Button1.”后，此时用户可能记不清所要用的属性名了，此时 Delphi 就弹出了提示窗来提供帮助，如图 1-11 所示。

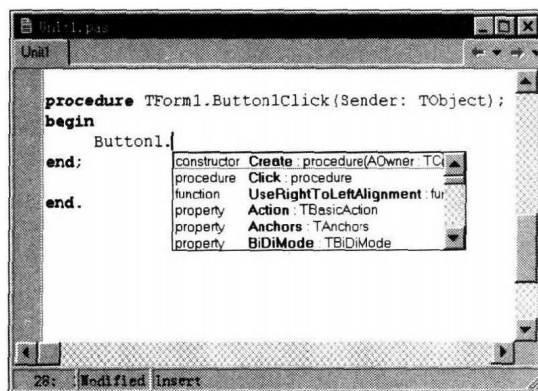


图 1-11 代码完成向导

2. 语句模板向导

该功能用于提供语句模板帮助用户完成语句的输入。例如，用户可以在键入“if”之后，按下 **Ctrl+J** 键，Delphi 就会弹出有关 if 的语句模板供用户选用，如图 1-12 所示。使用该功能有助于减少语法错误并统一语句格式。

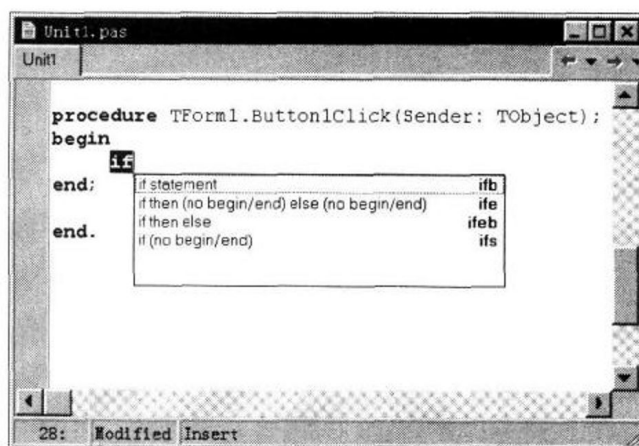


图 1-12 语句模板向导

3. 方法参数向导

该功能用于提示一个函数或过程所带的参数。有的函数或过程带有多个参数，这个功能就十分有用了。当用户键入函数名或过程名并键入左括号，稍作停顿，Delphi 就会将参数的提示显示在函数名或过程名的下方，如图 1-13 所示。



图 1-13 方法参数向导

1.2 如何建立 Delphi 程序

在了解了 Delphi 的基本开发环境后，下面就来学习如何建立一个 Delphi 程序。在这节中，我们将通过一个简单的例子向大家介绍开发 Delphi 程序的基本步骤。