国家技能型紧缺人才培养培训工程
高职高专软件技术专业规划教材

Delphi 实用编程技术

张卫东　主编

机械工业出版社
CHINA MACHINE PRESS

国家技能型紧缺人才培养培训工程
高职高专软件技术专业规划教材

Delphi 实用编程技术

主　编　张卫东
副主编　李　光
参　编　刘淑艳
　　刘　斌
　　齐丽君
主　审　王　伟

机械工业出版社

本书是根据教育部高等职业技术教育计算机专业 Delphi 编程技术课程的基本要求编写的。内容包括：Delphi 基础知识、Object Pascal 语言基础、窗体与界面设计、组件与程序交互、文件管理、图形图像处理、多媒体技术、数据库操作、Delphi 高级应用技术。本书简明扼要，内容全面，结构清晰，重点在编程，突出实用性，每节有应用实例，每章后有练习题。

本书为高等职业技术教育计算机专业 Delphi 编程技术课程教材，也适用于信息类其他专业，还可作为成人教育 Delphi 编程技术课程教材。

图书在版编目（CIP）数据

Delphi 实用编程技术/张卫东主编．—北京：机械工业出版社，2005.11

国家技能型紧缺人才培养培训工程．高职高专软件技术专业规划教材

ISBN 7-111-17874-2

Ⅰ.D… Ⅱ.张… Ⅲ.软件工具-程序设计-高等学校：技术学校-教材 Ⅳ.TP311.56

中国版本图书馆 CIP 数据核字（2005）第 132291 号

机械工业出版社（北京市百万庄大街 22 号 邮政编码 100037）

责任编辑：王玉鑫 版式设计：霍永明 责任校对：吴美英

封面设计：鞠 杨 责任印制：洪汉军

北京京丰印刷厂印刷

2006 年 1 月第 1 版·第 1 次印刷

787mm×1092mm 1/16·16.25 印张·402 千字

0 001—4 000 册

定价：24.00 元

凡购本书，如有缺页、倒页、脱页，由本社发行部调换

本社购书热线电话（010）68326294

封面无防伪标均为盗版

前　言

为适应高等职业技术教育的发展，满足计算机应用及软件技术专业技能型紧缺人才的培养要求，机械行业高职计算机专业教学指导委员会组成了“国家技能型紧缺人才培养培训教材编写组”，编写了一套具有高职特色的计算机专业系列教材，本书即是其中之一。

本书主要内容：Delphi 基础知识、Object Passcal 语言基础、窗体与界面设计、组件与程序交互、文件管理、图形图像处理、多媒体技术、数据库操作、Delphi 高级应用技术。本书内容符合教育部高等职业技术教育计算机应用及软件技术专业 Delphi 课程基本要求。按照“以实用为目的、加强编程训练”的原则，以“了解组件、学会编程、掌握实例”为教学目标，紧扣高职计算机专业技能型紧缺人才培养要求。淡化概念，突出编程方法与应用，简明扼要，结构清晰。在编写实例上有一定创新，每章的实例既方便教学，又有较强的实用性。

本书由沈阳职业技术学院软件学院张卫东副教授任主编，沈阳师范大学美术与设计学院李光任副主编，参加本书编写的有四川工程职业技术学院刘淑艳，沈阳职业技术学院刘斌，沈阳师范大学职业技术学院齐丽君，全书由张卫东、李光统稿，由河南工业职业技术学院王伟副教授主审。

本书在编写和出版过程中，参阅了有关书籍和教材，并得到了机械行业高职计算机专业教学指导委员会全体委员的帮助和支持。本书编写人员在此向参考书籍的作者表示衷心的感谢，向给予本书编者帮助、支持和指导的领导和同志们表示衷心的感谢，向本书的主审河南工业职业技术学院王伟副教授表示衷心的感谢。

因本书编写者的水平有限，书中难免存在疏漏之处甚或错误，恳请同行及读者提出批评和指正。

张卫东

于沈阳

目　录

第1章 基础知识

Delphi是Borland公司推出的可视化开发工具，它拥有世界上最快的编译器，并提供了丰富的组件集、强大的代码自动生成功能和丰富的数据库管理工具等。使用它的集成开发环境，编程人员可以更快地建立应用程序。

Delphi的英文原义是古希腊一个城市的名称，因她拥有阿波罗（Apollo）神殿而闻名于世。古希腊人认为Delphi是世界的中心，Borland公司将其可视化编程工具命名为Delphi，是期望它将成为可视化开发工具的先驱与核心。

1.1 认识Delphi

Delphi目前有1.0版、2.0版、3.0版、4.0版、5.0版、6.0版、7.0版等版本。其中的1.0版是在Windows 3.x下运行，Windows 95/98/2000推出后，Borland公司又陆续推出了可运行在Windows95/98/2000和WindowsNT上的版本，使用方便、功能强大的32位的2.0版、3.0版、4.0版、5.0版、6.0版和7.0版。

1.1.1 Delphi的主要特点

Delphi是基于Object Pascal语言的面向对象的开发工具，使用它的集成开发环境（IDE）可以快速地建立应用程序，许多传统的、常规的编程都可以借助于类库（Class Library）来实现。使用Delphi，既可以开发本地（Local）类型的软件，又可以开发客户/服务器（C/S）类型的软件。Delphi提供了丰富的组件集和强大的代码生成功能，使用Delphi开发的应用程序的用户界面和程序的健壮性完全可以和商业软件相媲美。

Delphi提供了丰富的数据库管理工具，它集成有Borland公司的数据库引擎BDE（Borland Database Engine）。借助于BDE，Delphi可与dBASE、Paradox以及支持ODBC的数据库链接。

Delphi的主要特点体现在以下几个方面：

1）Delphi为32位应用程序，因此其性能就像装上了涡轮引擎一样强劲有力，使用它可开发出功能强大的应用程序。

2）Delphi的编译器是目前世界上最快的32位本地代码（源代码）编译器。使用这种编译器产生的.EXE。运行文件是独立的，不需要链接运行时的解释器DLL。

3）Delphi可充分发挥Windows95/98/2000和WindowsNT的强大功能。由于Delphi可运行在Windows95/98/2000和WindowsNT之上，因此用户可获得无限制的虚拟内存。Delphi支持多线程、Unicode、MAPI、长文件名等Windows 95/98/2000和WindowsNT应用程序接口（API）。利用Delphi提供的一整套Windows95/98/2000组件，如目录浏览、状态栏、目录查看等，可建立完全符合Windows 95/98/2000的应用程序。完整的OLE自动控制和服务支持，使Delphi程序可以与其他支持OLE的程序相联系，如Visual Basic、Excel等。Delphi可完整地支持OCX技术，用户可以将OCX控制用拖放方式直接从组件选项板上加入到应用程序中。

4）Delphi 提供了多种 32 位可视组件库。在建立应用程序时，只需要拖放就能建立用其他工具很难完成的、优雅的、Windows 风格的应用程序。Borland 公司还将完整的可视组件库的源代码免费提供用户，这就使得重用和定制成为可能。

5）Delphi 是一种面向对象的程序设计语言，因此 Delphi 可以做到可视窗体的继承。也就是说，可以在一个已经存在的窗体基础上，利用继承的方式创建多个子窗体。子窗体具有父窗体的属性，同时也可拥有自己的独特属性。这就使生成的应用程序逻辑更为清楚、代码更易于管理和维护，从而大大地提高了效率。

6）Delphi 采用三层数据管理模式（数据层、对象层、应用程序层），把诸如数据模型、业务规则、窗体、对象等集中存储在对象存储库中。对象存储库中的组件可以与上层的应用程序无关，也可以与下层的数据源无关。当用户需要建立新的应用程序时，便可利用对象存储库中已经测试过的组件。这种最大限度的代码重用和方便的维护工作，无疑大大地提高了生产效率。

7）应用程序通过在 Delphi 中使用 Borland 公司提供的数据库引擎（BDE）功能，可以毫无障碍地使用多种数据库，不论是大型数据库还是 PC 中的数据库，如 Oracle、Sybase、Informix、INTERBASE、Microsoft SQL Server、Paradox、xBASE 系列数据库。

8）Delphi 中的数据感知功能，可使用户在开发应用程序时就可看到数据库的动态变化。

9）使用 Delphi 提供的数据库浏览器，在 Delphi 的集成开发环境中，可以浏览、修改、索引数据库。数据字典可在窗体和应用程序之间重用，从而允许用户快速地建立和维护数据的完整性，而不用另外编程。

10）为了便于维护程序，Delphi 将数据访问与业务规则从程序中分离出来，集中存储在数据模型对象中。当业务规则需要修改时，只需在数据模型集进行修改。程序运行调用这些数据模型时，修改的结果会自动反映在应用程序中。

此外，Delphi 为用户免费提供了两个用户许可的本地 INTERBASE（数据库管理系统）。

1.1.2 Delphi 的集成开发环境

Delphi 是新一代的 32 位快速应用程序开发工具（RAD）。利用它可以开发基于 Windows 的 32 位应用程序。Delphi 的全部工具箱中包括 Delphi 集成开发环境（IDE）、数据库桌面、数据库资源管理器、图形编辑器、INTERBASE 数据库等。开发应用程序可在 Delphi 的集成开发环境中一次完成。

进入 Delphi 后，首先看到四个前景窗口与一个隐藏在后的窗口，如图 1-1 所示，它们为：

- 主窗口：位于屏幕的顶部，即标题为 Delphi7-Project1 的窗口，它包括菜单、工具栏、组件选项板（Component Palette）等。
- 窗体（Form）窗口：位于屏幕的右边，即标题为 Form1 的窗口。
- 对象编辑器：位于屏幕的左下部，即标题为 Object Inspector 的窗口。
- 对象结构浏览：位于屏幕的左上部，即标题为 Object TreeView 的窗口。
- 代码编辑器（Code Editor）：位于窗体窗口的后面，即标题为 Unit1.pas 的窗口。

1. 主窗口

主窗口中包含菜单栏、工具栏及组件选项板。

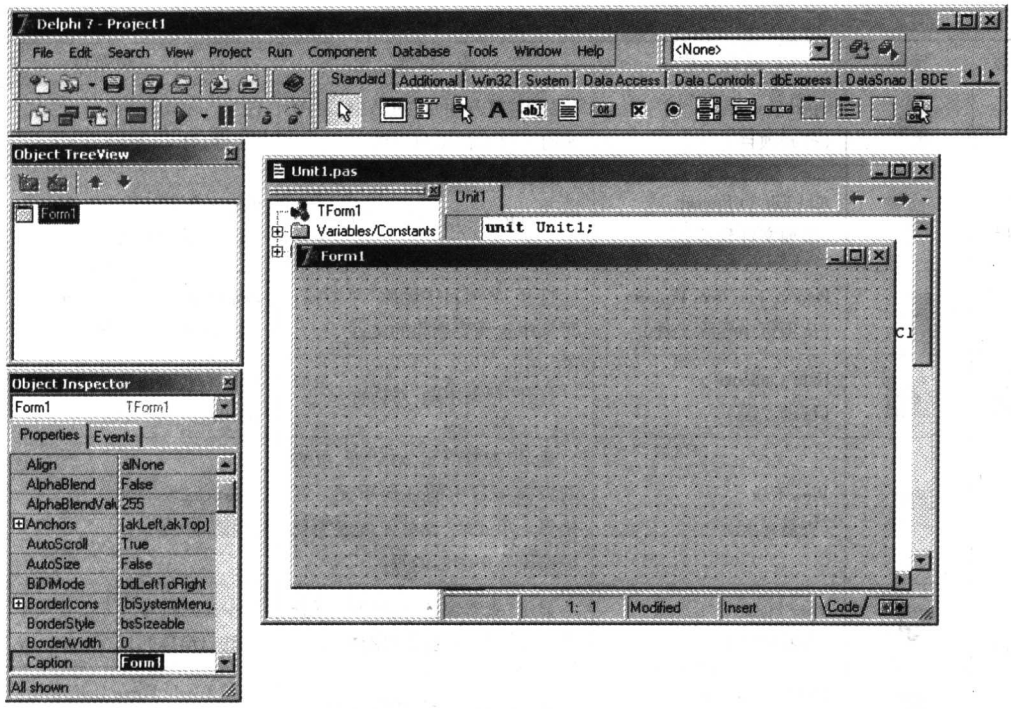

图 1-1　Delphi 的集成开发环境

(1) 菜单栏　菜单栏中共有 11 个菜单项，分别为 File（文件）、Edit（编辑）、Search（查找）、View（视图）、Project（项目）、Run（运行）、Component（组件）、Database（数据库）、Tools（工具）、Window（窗口）和 Help（帮助）。

Delphi 的菜单可根据当前的使用状态，增加或取消一些菜单选项。用户还可通过菜单将更多的工具添加到开发环境中来。

(2) 工具栏　Delphi 的工具栏提供对菜单命令的快捷存取方式。当光标移动到其上时就会看到关于该图标的提示。Delphi 中的工具栏包括 Standard（标准工具栏）、View（查看工具栏）、Debug（调试工具栏）、Custom（定制工具栏）、Desktops（桌面工具栏）、Component Palette（组件选项板）、Internet（互联网工具栏）等。工具栏中的按钮及其功能见表 1-1。

表 1-1　默认的工具栏按钮及功能

工具栏	按钮	按钮名称	功　能	等价菜单
标准工具栏		New (新建)	打开“新建项目”对话框	File|New|Other (文件|新建|其他)
		open (打开)	显示“打开”对话框，可用来打开已有文件	File|Open (文件|打开)
		Save (保存)	对当前打开项目中所有修改过的文件以原来的文件名保存	File|Save (文件|保存)
		Save all (全部保存)	保存所有打开的文件，包括当前的项目及模块	File|Save|All (文件|全部保存)

（续）

工具栏	按钮	按钮名称	功　能	等价菜单
标准工具栏		Open Project （打开项目）	显示“打开项目”对话框，可用来打开已有的项目文件	File|Open|Project （文件|打开项目）
		Add File to Project （添加文件到项目）	打开“添加到项目”对话框，可将已有的单元及相关的窗体添加到 Delphi 的项目中	Project Add to Project （文件|添加到项目）
		Pemove File from Project （从项目中删除文件）	打开“从项目中删除”对话框，可从中选择要删除的文件	Project|Remove from Prject （文件|从项目中删除）
定制工具栏		Help Contents （帮助主题）	打开“帮助主题”对话框	Help|Delphi Help （帮助|Delphi 帮助）
查看工具栏		View Unit （查看单元）	显示“查看单元”对话框，可查看当前项目中的任何单元。在选择一单元后，该单元就成为代码编辑器中激活的页	View|Unit （视图|单元）
		View Form （查看窗体）	显示“查看窗体”对话框，可查看当前项目中的任何窗体	View|Form （视图|窗体）
		Toggle Form/Unit （切换窗体/单元）	在窗体和单元间进行切换	View Toggle Form/Unit （视图|切换窗体/单元）
		New Form （新建窗体）	建立一空的窗体和新的单元，并将它们加入到项目中	File|New|Form （文件|新建窗体）
调试工具栏		Run （运行）	编译并执行应用程序	Run|Run （运行|运行）
		Pause （暂停）	暂停程序的执行，并将执行的位置指向程序中的下一行	Run|Program Pause （运行|程序暂停）
		Trace Into （追踪）	执行执行点高亮的语句。追踪可一次执行一条语句。当子例程从调用的调试位置返回时，下一执行语句为子例程调用的下一可执行语句	Run|Trace Into （运行|追踪）
		Step Over （单步）	一次执行一条语句，而不转到子例程中。子例程作为一个语句来考虑，下一执行语句为调用子例程的下一可执行语句	Run|Step Over （运行|单步）
桌面工具栏		Desktop Speedsetting 桌面设置	选择保存的桌面布局	
		Save current desktop 保存当前桌面	显示“保存桌面”对话框，可用来保存当前桌面设置	
		Set debug desktop 设置调试桌面	设置调试桌面	

（3）组件选项板　工具栏的右边为组件选项板。其上方是组件的分类页标签，用鼠标单击不同的页标签时，下方就会出现不同内容的组件。

在 Delphi 中，组件具有举足轻重的作用。Delphi 可以快速地开发应用程序，一个决定性的因素是使用了组件。组件是 Delphi 程序的构造块，它可方便地“插入”到应用程序中，帮助开发者快速地制作出良好的、与 Windows 界面类似的界面。例如，要在界面中制作一个“确定”按钮，这时只要从组件选项板中选择按钮组件，用鼠标将按钮拖放到窗体上，同时调整好其位置和大小，配置好相应的属性（大部分可使用默认的属性），并在事件中定义该按钮要执行的具体操作即可，这样一个按钮就制作好了。这在以前，要制作出如此简单的按钮，需要编写的程序代码是相当多的。

Delphi 的组件分为可见组件（如文本编辑框、对话框等）和不可见组件（如系统定时器等）。

Delphi将不同的组件按功能的不同，放在不同的标签上，如 Standard、Additional、Win32、System、DataAccess、DataControls、dbExpress、DataSnap、BDE、ADO、INTERBASE、WebServices、Internet Express、Internet、WebSnap、FastNet、Decision Cube、QReport、Dialogs、Win3.1、Samples、ActiveX、Com + 、Indy Clients、Indy Services、Indy Misc、Servers 等，使用户一目了然，快速方便地找到自己需要的组件。在 Delphi 中，用户可以根据自己的需要创建自己的组件，将它们放到组件选项板上供以后使用，由此可见 Delphi 具有很强的延展性及灵活性。

（4）对象结构浏览

在对象结构浏览（见图 1-2）中显示出放置在窗体、数据模块或框架上的可见的及不可见的组件关系结构图，同时在该窗口中，通过拖动也可以创建组件之间的关系。

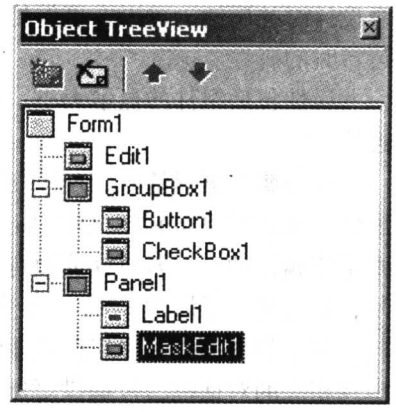

图 1-2　对象结构浏览

2. 窗体

对于最终用户来说，窗体仅仅是一个窗口。在 Delphi 中，窗体是接受组件（在设计时由程序设计人员放置的或在运行时用代码动态生成的）的一个窗口，而不管运行时该窗口的作用如何。因此，窗体就是设计程序的工作底稿区或画布或搭积木的底座，在窗体中放置适当的组件，并安排好相互间的位置，整个程序的界面就算完成了。接着再做组件属性的设置与编写相关的事件处理程序，这样就完成了整个程序的设计。

当然也可将窗体理解为存放其他组件的一个特殊组件。窗体的外部特征与 Windows 95/98/2000/XP/NT 的窗口是类似的。窗体是 Delphi 应用程序的焦点，无论是对窗体添加组件、编辑属性或编写代码，都是在编辑窗体。

窗体保存在两个独立的文件中。

.DFM 文件保存的是窗体的二进制图形文件。任何对窗体的可视化特性编辑，如改变高度、颜色、边界等，都保存在 .DFM 文件中。Delphi 提供了一个工具软件，可用来以文本方式打开窗体文件并显示在文本编辑器中，其方法是在窗体单击鼠标右键，在弹出的快捷菜单中选择“ViewasText”菜单命令即可。

单元文件 .PAS 保存的是 .DFM 文件的源代码。在 .PAS 文件中可以编写事件处理程序。

当建立和修改窗体时，Delphi 将同步保存 .DFM 文件和源代码文件。在添加新的窗体时，Delphi 将建立与之相关的单元文件，并将其添加到项目文件中的 use 子句中。

在编译时，若出现错误，则 Delphi 在代码编辑器的消息框中显示错误或高亮出错的行。

3. 对象编辑器

对象编辑器是组成应用程序的外观与代码的通道。使用对象编辑器可以设置放在窗体上的组件（或窗体本身）的属性及通过事件处理程序来帮助用户建立代码。

对象编辑器如图 1-3 所示。

对象编辑器顶部的对象选择器为下拉式列表框，其中包含有当前激活的窗体中的所有组件并显示出它们的组件类型。从而可以在当前的窗体中快速地选择不同的组件。

对象编辑器有两个页标签：属性页标签及事件页标签。

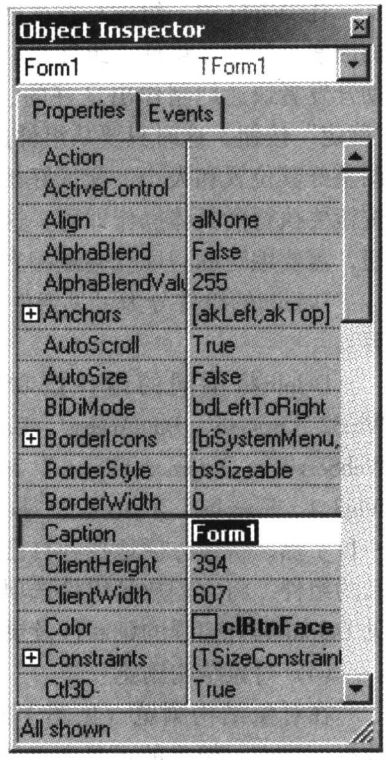

图 1-3　对象编辑器

（1）对象编辑器的属性页（Properties）标签　对象编辑器的属性页标签可用来观察、设置窗体上组件或窗体本身设计时的属性。可以在事件处理程序中编写代码来设置运行时的属性。

属性包括组件的显示大小、标题、活动状态、可见状态等。

选择属性页标签时，左边列出了所有设计时可修改的属性（就是对象类别定义中 Published 区段的属性），右边则是该属性的值。若属性前面有“+”号，则表示该属性中还有子属性，单击这里的“+”号，就会列出子属性，“+”号变为“-”号；再单击“-”号就会折叠属性。另外，有些较为复杂的属性，单击属性或属性值字段时，属性值字段右边会出现“…”，单击它，就会出现相应的对话框，可提供更进一步的信息。

属性页中只显示窗体中选定组件的属性。

在设计时设置的属性可以定义组件的初始状态。

（2）对象编辑器的事件页（Events）标签　事件标签则列出组件将做出反应的各种事件，如单击事件、按回车键事件等，由开发者决定组件对某种事件将采取何种操作，其方式就是在相应的事件中填写发生该事件后将执行的程序或过程名。

对象编辑器的事件页标签可将窗体和组件连接到程序事件中。当单击事件页标签时，Delphi 生成一事件处理程序，并将焦点转换到代码编辑器中。使用代码编辑器，可以在事件处理程序中编写组件或窗体对特殊事件处理的代码。

事件页中只显示窗体中选定组件的事件。

4. 代码编辑器

代码编辑器是一具有全部编辑特征的编辑器。可用作编辑应用程序的代码，其强大的功能特征包括：

- 简捷的编辑。

- 带颜色的语法高亮。
- 多个及成组的撤消编辑。
- 具有完全可使用的编辑命令。

将光标放在代码编辑器的符号上，按“F1”键，可以获得该符号的语法帮助。

打开一新的项目后，Delphi 对主窗体中的单元在代码编辑器中添加一页标签。

(1) 单元（Unit） 单元为一独立的可编译的代码模块，其中包含公共部分（界面部分）和局部部分（实现部分）。

Delphi 中的每一窗体都有与之相联系的单元。

单元的源代码保存在 .PAS 文件中，编译后的文件为 .DCU，连接 .DCU 文件后的执行文件为单个的 .EXE 或 .DLL 文件。

(2) 事件处理程序 窗体方法依附在事件上，当指定的事件发生时，就执行相应的事件处理程序。

当使用对象编辑器指定代码到组件事件中时，Delphi 生成过程头及 begin…end 块。如对按钮单击事件，Delphi 生成的代码为：

```
procedure TForm1.Button1Click (Sender: TObject);
begin
…
end;
```

单击“Button1”按钮后要执行的代码写在 begin…end 块之间。

1.2 Delphi 程序

1.2.1 程序的基本结构

Delphi 的集成开发环境是通过项目（Project）的方式来组织和管理应用程序开发过程中的各类文件的。

(1) 项目与项目文件 开发应用程序的所有相关文件都组织在一个项目中，有些是在程序设计阶段产生的，如项目文件（.DPR）、窗体文件、单元文件等；有些是在项目编译和连接过程中生成的，如单元目标文件（.DUC）等。Delphi 项目中包含的文件很多，但大部分文件是由 Delphi 自动创建和维护的，对开发者来说只需关注单元文件和窗体文件。

项目文件是 Object Pascal 语言源代码文件，其中列出了项目中所包含的全部单元文件名，如窗体、单元等。项目文件是由 Delphi 自动维护的，不必人工修改。

(2) 项目文件与单元文件的关系 项目文件是特殊的单元文件，可理解为主单元文件，或者说主程序，而其他的单元文件可以看作是被项目文件所调用的子程序。

当编译一个项目时，Delphi 编译器首先查看项目文件，然后编译项目文件中列出的各个单元文件，最后把这些文件连接成可执行文件。

(3) 窗体文件 窗体在设计阶段可用来放置各种 VCL 组件，在运行阶段是与用户交互的界面。窗体中的所有信息保存在两个同名（扩展名不同）的文件中。一个是窗体文件（窗体定义文件）.DFM；另一个是每个窗体对应的单元文件 .PAS。

窗体文件用来保存窗体及其上对象的特征，如大小、位置、颜色、显示方式等。该文件是二进制代码文件，无需用户进行修改。若要查看的话，可以用鼠标右键单击窗体上除标题栏以外的任何位置，在弹出的快捷菜单中选择“View As Text”菜单命令，就可以看到该窗体文件的文本形式。

(4) 单元文件　单元文件用来保存窗体及窗体上组件的其他信息，主要是窗体或组件的事件处理代码。

Delphi 使用的是 Object Pascal 语言，因此 Delphi 生成的程序都采用 Object Pascal 的语法。

Delphi 的每个窗体都有一个对应的单元文件，单元文件中包含了窗体和窗体上组件的事件处理程序，扩展名为 .PAS。

下面以一个例子来说明，Delphi 程序是如何构成的。该程序是在窗体中添加一个下拉式菜单组件后，代码编辑器中的一段源程序（单元）：

```
unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms, Dialogs;
Type
  TForm1 = class (TForm)
  MainMenu1: TMainMenu;
Private
  {Private declarations}
Public
  {Public declarations}
end;
var
  Form1: TForm1;
Implementation
{$R *.dfm}
end.
```

单元文件中包含有：单元标题、接口部分和实现部分。

单元中可以包含事件处理程序、过程及相关的函数。

一个单元由以下五大部分组成。

(1) 单元标题（Unit Heading）　单元文件的第一句话。

单元标题指定单元的名称。这里的名称可在引用该单元时的 uses 子句中使用。名称必须是惟一的，也就是说同时不能使用相同的单元名字。

单元标题不要在代码编辑器中随意进行修改，否则就会出现编译错误。

(2) 界面部分（Interface Part）　在标题头后，以关键字 interface 开始，implementation 前的部分。界面部分说明公共的常量、类型、变量、过程及函数等。对过程和函数，只列出说明部分，过程体或函数体在实现部分。

其中 uses 子句确定了本单元中使用了哪些其他单元，一般来说，Delphi 将一些常用的标

准单元，如 Windows、SysUtils、Classes、Graphics、Forms 等单元自动加入到该子句中。当然也可以加入自己的非标准单元。

(3) 实现部分（Implementation Part） 从关键字 implementation 开始直到本单元末尾或初始化部分之前。主要用于定义在界面部分说明的全部公共过程体和函数体。也可说明局部的常量、类型、变量、过程或函数。

其中的 {$R * .DFM} 是编译指令，指示编译器在编译时要连接窗体。

(4) 初始化部分（Initialization Part） 为可选的部分，以关键字 initialization 开始，位于实现部分之后，主要是对本单元中的一些数据进行必要的初始化，如给变量赋初值、为实现部分分配资源等。

(5) 结束部分（Finalization Part） 为可选的部分，若单元具有初始化部分就必须具有结束部分。以关键字 finalization 开始，直到单元结束，主要为结束单元的语句。在初始化部分获得的任何资源（内存、文件等）在结束部分都要释放。结束部分的执行顺序与初始化部分的执行顺序是正好相反的。

1.2.2 Delphi 程序设计过程

Delphi 程序设计是可视化的，也就是说在程序设计阶段看到的程序界面与最终的运行结果非常相似。在 Delphi 中开发应用程序是作为一个项目处理的，其主要工作有：设计窗体、用代码把窗体与窗体、窗体与功能连接起来。

在 Delphi 中创建应用程序的过程一般如下：

1）使用“File | New | Application”菜单命令创建新的项目和窗体。在 Delphi 中项目是构成应用程序或动态连接库的所有文件（窗体、单元、资源等）的集合。

2）在窗体中加入组件。在组件选项板中选定需要的组件，然后在窗体的适当位置单击。接着可以对窗体和窗体上的组件进行适当地安排。

3）设置窗体和组件的属性。在对象编辑器中设置的属性是静态属性，在程序中设置的属性是动态属性。

4）编写事件处理程序。

5）保存项目文件和单元文件。保存时最好将不同的项目保存在不同的文件夹中。

6）编译、调试和运行程序。选择“Run | Run”运行应用程序。若出现问题或程序的运行结果达不到预期的结果，还可以使用 Delphi 提供的调试器来调试、修改，直到满意为止。

1.2.3 一个简单的 Delphi 程序

现在来设计第一个程序，在窗体上生成一个按钮，当用鼠标单击时，就结束这个程序。通过这个简单的程序进一步体会一下 Delphi 程序的设计过程。其操作过程如下。

1）进入 Delphi 的集成开发环境。

2）单击对象编辑器窗口中的“Caption（标题）属性”，将属性值中显示的“Form1”改为“欢迎使用 Delphi”。这时窗体的标题就显示“欢迎使用 Delphi”。

3）单击组件选项板标准页标签上的“OK”按钮：然后在窗体的中央拖放生成一按钮。这时按钮上的标题为“Button1”。

4）单击对象查看器窗口中的“Caption（标题）属性”，将属性值中显示的“Button1”改

为“退出”。这时按钮上的文字就显示“退出”。

5）单击对象编辑器窗口中的“Font（字体）属性”，再单击属性值右边的“…”，弹出字体对话框。

6）在字体对话框中选定隶书、字体样式选定为规则的、大小选定为四号字。这时窗体中的按钮上的“退出”二字就变为四号字规则的隶书。

7）使用代码编辑器建立代码。双击“退出”按钮，弹出生成的相对于按下这个按钮的事件处理函数。只要填入想要做的操作就完成了程序设计。对这里的“退出”按钮来说，就是加入 close 命令。即填入语句：

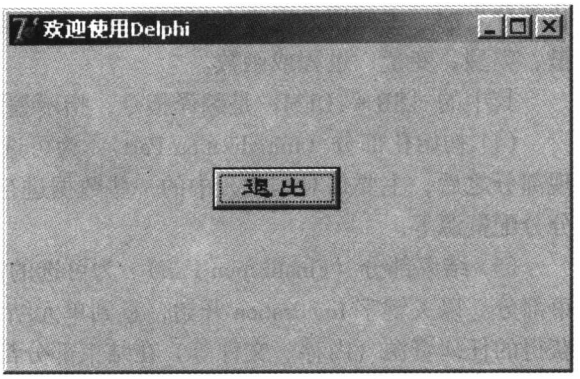

图 1-4　退出按钮的运行结果

```
Close;
```

8）运行程序。选择“Run | Run”菜单命令，经编译、连接后，运行的结果如图 1-4 所示。

当单击“退出”按钮时，就退出程序的执行。

习　题　1

1. Delphi 的集成开发环境由哪几部分组成？各部分的功能是怎样的？
2. 怎样修改窗体或组件的属性？
3. 一个 Delphi 应用程序的源文件有哪些？运行后又生成了哪些文件？各文件的含义是什么？