



设计与开发指导丛书

Delphi 7

数据库系统设计与开发



张春林 马成勇 刘均 编著



清华大学出版社

► 软件设计与开发指导丛书

Delphi 7 数据库系统设计与开发

张春林 马成勇 刘均 编著

清华大学出版社

北 京

内 容 提 要

Delphi 是一个极为优秀的快速应用程序开发 (RAD) 工具, 在开发数据库方面功能尤其强大。本书以实例教学的方式全面系统地讲述了 Delphi 开发数据库的编程方法, 深入探讨了 Delphi 7 的最新技术。

本书共 7 章, 主要包括 Delphi 程序设计基础知识和 6 个具有代表性的实际应用程序的详细开发过程。本书有机地将 Delphi 开发应用程序的理论和实践结合在一起, 详细、深入地介绍了使用 Delphi 开发应用程序 (特别是应用面最广的管理信息系统) 的整个流程, 引导读者开发实用的、功能全面的应用程序。

本书内容新颖、实用, 叙述浅显易懂, 可作为大专院校相关专业学生的学习用书和培训班的教材, 亦可供 Delphi 程序设计人员及数据库开发人员参考。

版权所有, 盗版必究。

本书封面贴有清华大学出版社激光防伪标签, 无标签者不得销售。

图书在版编目 (CIP) 数据

Delphi 7 数据库系统设计与开发/张春林, 马成勇, 刘均编著.

—北京: 清华大学出版社, 2003

ISBN 7-302-07308-2

I. D… II. ①张… ②马… ③刘… III. 软件工具—程序设计

IV. TP311.56

中国版本图书馆 CIP 数据核字 (2003) 第 085845 号

出 版 者: 清华大学出版社

地 址: 北京清华大学学研大厦

<http://www.tup.com.cn>

邮 编: 100084

社总机: 010-62770175

客户服务: 010-62776969

组稿编辑: 科 海

文稿编辑: 安 靖

封面设计: 付剑飞

版式设计: 卞雨桂

印 刷 者: 北京科普瑞印刷有限责任公司

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印张: 19.75 字数: 480 千字

版 次: 2003 年 10 月第 1 版 2003 年 11 月第 2 次印刷

书 号: ISBN 7-302-07308-2/TP·5306

印 数: 5001~8000

定 价: 29.00 元

丛 书 序

计 算机软件产业是当今世界上发展最快的朝阳产业。20 世纪 90 年代以来，世界软件市场年增长率都在 15% 以上。进入 21 世纪，全球软件产业每年的总产值达到 5000 多亿美元，占当年信息产业总产值的 60% 以上。据预测，今后几年内，软件市场的年增长率将提高到 17.3%。在软件产业发达的美国，支持 IT 经济的骨干力量是 5000 多家软件公司。2000 年以来，软件业超过汽车制造业成为美国最大的产业，并创造了 343.5 万个工作岗位，相当于美国劳动力总数的 3%。

20 世纪 90 年代以来，随着国内软件市场的不断扩大，我国软件产业逐步走上较为成熟与稳定的发展之路，软件产业发展呈现以下趋势：软件产业发展环境大大改善；信息化成为软件产业的大市场；软件产业上规模、上档次，成为中国经济发展的主导产业。

随着我国软件产业的高速发展，软件人才日益供不应求，而且需求量呈快速递增态势，但目前我国软件人才的数量、质量和结构离业内需求尚有很大差距。软件产业是一个智力、科技、文化密集的行业，软件人才的培养也必须本着这样一个原则和目标，建立科学全面的培养体系，才能造就一支高素质、复合型、具有持续发展能力的软件人才队伍。

这两年，IT 行业中最令人关注的职位当属“软件工程师”了，其需求量在 IT 人才市场中总是名列第一，成为 IT 人求职的热点。软件工程师的未来是什么？如何才能成为一名合格的软件工程师？从软件技术的发展及软件产业的竞争来看，只有不断学习，软件工程师才能应对未来的各种挑战。

为了满足社会对软件人才（尤其是软件工程师）的需要，让更多的人可以更快地学到实用的软件理论、技术与方法，帮助广大开发人员迅速提高软件设计与开发的能力，北京科海培训中心以实用性和工程实践性为导向，精心策划并组织专家编写了“软件设计与开发指导丛书”。丛书涵盖了流行的软件开发语言、编程工具、数据库开发技术、网络编程和多媒体开发技术。各书的作者都是长期从事软件设计与开发的从业人员，具有丰富的实践经验和编程技巧，相信他们的智慧结晶会为我国软件人才的培养起到促进作用。

丛书以培养面向 21 世纪计算机专业应用人才（以软件工程师为主）为目标，以简明实用、便于学习、反映计算机技术最新发展和应用为特色。丛书分数据库系列、网络编程系列、多媒体应用开发系列，各系列从结构设计、内容安排到体例都经过精心设计，具有以下特点：

- 内容适度，讲解鞭辟入里，便于自学。本丛书不仅面向广大软件从业人员及普通高等院校学生，更立足于希望通过基础程序设计训练，继而成为高级软件人才的有志青年。因此，在叙述和内容安排上尽量通俗易懂，力求

把握软件设计思想的来龙去脉，把软件设计的全过程讲透。

- 强化软件设计与开发的实现过程。要想成为一名出色的软件工程师，必须深入领会软件设计的奥秘，惟有这样才能做到得心应手。本丛书不像一般的软件系统使用手册那样仅列出各种语法，也不像一般的规范教科书那样只注重于抽象的理论方法，而是经过仔细遴选，从软件实现的角度精心设计出一系列富有代表性的应用案例，并给予详细的解析。
- 理论结合实践。计算机是一门实践性很强的科学，丛书贯彻从实践中来到实践中去的原则，把抽象的技术理论与极具实践指导意义的实例结合起来，便于学习和理解，是学生进行相关课程毕业设计的好帮手。
- 丛书自成一个完整的体系。每本书既相对独立，又相互衔接和呼应，为总的培养目标服务。读者可以依自己的兴趣特长选择其中的一本或多本进行研习。

丛书中的每一本的示例代码读者都可以在科海网站上下载：<http://www.khp.com.cn>。

党的“十六大”提出了“以信息化带动工业化”的发展战略，指明了我国信息界前进的方向。从产业发展的高度上看，伴随着优秀软件人才的出炉，中国软件业必将有“直挂云帆济沧海”的一日！

丛书编委会

2003年8月

前 言

Delphi是Borland公司推出的一个基于Windows的、以Object Pascal语言为核心的、快速开发应用程序的可视化平台，是当今最流行的Windows软件开发工具之一。

Delphi提供了灵活的可视化设计工具，将可视化界面与面向对象技术完美、紧密地结合起来，并封装了Windows编程的复杂性。它的优势之一是开发数据库应用程序。在这方面，它不仅提供了大量的数据库组件，并配有数据库引擎BDE，能通过SQL Links和ODBC等访问多种数据库，同时提供了强大的开发网络数据库的能力。

本书不仅介绍了Delphi 7的基础知识，还介绍了6个完整的Delphi应用程序的开发实例，这几个实例具有很强的实用性和代表性。

全书共7章。

- 第1章，Delphi基础，简要介绍了Delphi集成开发环境以及Delphi常用组件的使用。
- 第2章，房屋租赁管理系统（BDE），介绍了一个通过数据库引擎BDE访问数据的完整的桌面型数据库管理系统的开发。
- 第3章，图书管理信息系统（ADO），介绍了一个图书管理系统中的主干部分，对于数据库的访问是通过ADO来实现的。
- 第4章，企业生产管理系统（C/S），介绍了一个客户机/服务器模式的数据库管理系统的实现过程，包括在数据库服务器端（SQL Server 2000）数据表的设置和操作，以及客户端界面和程序设计的实现。
- 第5章，网上学生选课系统（B/S），详细介绍了Delphi开发Web Server应用程序的实现技术和技巧。
- 第6章，画图程序，画图程序是一个图形图像处理的应用程序，其功能和界面完全模仿Microsoft公司的Windows操作系统中提供的画图软件。
- 第7章，FTP客户端程序，通过实现互联网中文件的上传和下载，介绍了网络客户端应用程序的开发过程。

书中所有的应用程序均在Delphi 7开发环境中调试通过。

由于水平所限，书中难免存在错误和不足之外，敬请广大读者批评指正。作者的E-mail地址为zcl@263.net。

编 者
2003年9月

目 录

第1章 Delphi基础	1
1.1 Delphi简介	1
1.1.1 Delphi的历史	1
1.1.2 Delphi中的有关概念	1
1.2 Delphi 7的集成开发环境	2
1.3 Delphi中的文件及其结构	4
1.3.1 单元文件及其结构	5
1.3.2 窗体文件及其结构	7
1.3.3 工程文件及其结构	8
1.4 Delphi常用组件的使用	9
1.4.1 窗体与组件	10
1.4.2 按钮组件的使用	11
1.4.3 文本组件的使用	12
1.4.4 列表组件的使用	14
1.4.5 分类组件的使用	16
1.4.6 菜单组件、工具栏组件和状态条组件的使用	17
1.4.7 标准对话框组件的使用	22
1.4.8 时钟组件的使用	23
1.5 数据库组件的使用	24
1.5.1 BDE组件	24
1.5.2 TTable组件	28
1.5.3 TQuery组件	30
1.5.4 TDataSource组件	33
1.5.5 数据控制组件	34
1.5.6 与数据库有关的其他组件和方法	38
第2章 房屋租赁管理系统（BDE）	39
2.1 系统总体设计	39
2.2 系统数据库设计	40
2.2.1 初始化数据库桌面	40
2.2.2 创建数据表	41
2.3 工程总体设计	43
2.3.1 工程窗体结构设计	43
2.3.2 创建数据控件模块	44

2.4 用户登录窗体设计	46
2.4.1 用户登录窗体界面设计	46
2.4.2 用户登录窗体功能设计	46
2.5 主窗体设计	48
2.5.1 主窗体界面设计	48
2.5.2 主窗体功能设计	49
2.6 房产管理窗体设计	51
2.7 合同管理窗体设计	53
2.7.1 合同信息管理	53
2.7.2 到期合同处理	66
2.7.3 合同信息查询	72
2.8 租金管理窗体设计	77
2.8.1 租金收取	77
2.8.2 按合同号租金查询	80
2.9 信息统计窗体设计	82
2.10 系统维护窗体设计	86
2.10.1 数据备份	87
2.10.2 数据恢复	88
2.10.3 密码管理	90
第3章 图书管理信息系统 (ADO)	92
3.1 ADO组件简介	92
3.2 图书管理信息系统的实现	96
3.2.1 系统总体设计	96
3.2.2 系统数据库设计	96
3.2.3 工程窗体结构设计	98
3.2.4 主窗体设计	98
3.2.5 数据模块设计	100
3.2.6 图书管理窗体设计	101
3.2.7 读者管理窗体设计	106
3.2.8 借阅管理窗体设计	112
3.2.9 信息查询窗体设计	121
3.3 本章小结	129
第4章 企业生产管理系统 (C/S)	130
4.1 系统总体设计	131
4.2 系统数据库设计	132
4.2.1 安装MS SQL Server 2000	133
4.2.2 创建数据库	133
4.2.3 创建数据表和视图	138

4.2.4 为“系统功能”表加入初始数据	144
4.3 工程总体设计	144
4.3.1 工程窗体结构设计	144
4.3.2 创建公共使用单元	148
4.3.3 修改工程主文件	151
4.3.4 创建数据模块	152
4.4 数据库设置窗体设计	155
4.5 用户登录窗体设计	156
4.5.1 用户登录窗体界面设计	156
4.5.2 用户登录窗体功能设计	157
4.6 主窗体设计	161
4.6.1 主窗体界面设计	161
4.6.2 主窗体功能设计	163
4.7 用户及权限窗体设计	165
4.7.1 设计用户权限窗体用到的数据集	165
4.7.2 用户及权限窗体界面设计	166
4.7.3 用户及权限的窗体功能设计	167
4.8 员工信息窗体设计	178
4.8.1 设计员工信息窗体用到的数据集	178
4.8.2 员工信息窗体界面设计	180
4.8.3 员工信息窗体功能设计	182
4.9 团体汇报窗体设计	186
4.9.1 设计团体汇报窗体用到的数据集	186
4.9.2 团体汇报窗体界面设计	186
4.9.3 团体汇报窗体功能设计	188
4.10 各部门工作项目工时统计窗体设计	194
4.10.1 设计各部门工作项目工时统计窗体用到的数据集	194
4.10.2 各部门工作项目工时统计窗体界面设计	195
4.10.3 各部门工作项目工时统计窗体功能设计	196
4.11 本章小结	198
第5章 网上学生选课系统 (B/S)	199
5.1 B/S体系结构简介	199
5.2 Web Broker技术	200
5.2.1 Web Server应用程序类型	200
5.2.2 Web Broker技术	201
5.3 系统设计	202
5.4 系统数据库设计	204
5.5 建立页面模板	207

5.5.1 创建登录页面.....	207
5.5.2 创建选课页面.....	209
5.5.3 重修课程查询页面.....	213
5.6 程序设计.....	215
5.6.1 建立Web Server应用程序.....	215
5.6.2 编写事件处理程序.....	217
5.6.3 编译.....	231
5.7 应用程序的发布.....	232
5.8 本章小结.....	233
第6章 画图程序.....	234
6.1 Delphi中图形图像处理的实现.....	234
6.1.1 关于图形媒体的程序设计.....	234
6.1.2 关于图像媒体的程序设计.....	240
6.2 画图程序的实现.....	244
6.2.1 程序界面设计.....	244
6.2.2 程序初始化.....	248
6.2.3 绘制图形.....	249
6.2.4 菜单操作实现.....	259
6.3 本章小结.....	269
第7章 FTP客户端程序.....	270
7.1 FTP协议简介.....	270
7.2 FTP组件的使用.....	271
7.3 FTP客户端程序的实现.....	273
7.3.1 程序界面设计.....	273
7.3.2 程序初始化.....	279
7.3.3 实现“连接”按钮的OnClick事件.....	283
7.3.4 弹出式菜单PopupMenuLocal的实现.....	285
7.3.5 弹出式菜单PopupMenuRemote的实现.....	292
7.3.6 本地文件列表有关事件的实现.....	299
7.3.7 服务器文件列表有关事件的实现.....	302
7.4 本章小结.....	303

第1章 Delphi 基础

1.1 Delphi简介

1.1.1 Delphi的历史

Delphi是Borland公司(Inprise公司的前身)推出的一种可视化的、方便快捷的Windows应用程序开发工具,目前它已成为与Microsoft公司的Visual Basic (VB)、Visual C++ (VC)等齐名的开发工具。由于Delphi既具有VC的强大功能和高效性,同时又具备VB编程的方便性,因此使用Delphi开发程序的用户的数量正迅速地增长。

Delphi的基础语言是对象(Object) Pascal。对象Pascal是一种强类型语言,与其他语言相比,它提供了一个快速的编译器,优化的编译模式在很大程度上提高了代码质量。自从Delphi 1.0推出以来,相继出现了Delphi 2.0、3.0、4.0、5.0、6.0和7.0版本。各版本虽然向后兼容,但随着版本的提高,在许多方面都作了改进。2002年8月推出的Delphi 7是Delphi的最新版本。

1.1.2 Delphi中的有关概念

1. 面向对象程序设计

面向对象程序设计(Object-Oriented Programming, OOP)是Delphi开发程序的基础。它将人们认识世界过程中普遍采用的思维方法应用到程序设计中。面向对象的程序设计涉及到对象、封装、类、继承和多态等几个基本概念。

对象

对象是现实世界中存在的事物,它们可以有形的,如一张桌子、某个人等;也可以是无形的,如某个方案、一次图书的借阅关系等。对象是构成现实世界的一个独立单位,人们对世界的认识就是从分析对象的特征入手的。

类

类是具有共同性质的事物的集合。在面向对象的程序设计中,类是一个独立的程序单位,它有一个名字(类名),其内部包括成员变量,用于描述对象的属性;还包括类的成员方法,用于描述对象的行为。

类对象

类对象是类的实例化。类是一个抽象的概念,用类的方式来解决就必须用类创建一个实例化的类对象,然后通过类对象去访问类的成员变量和调用类的成员方法。一个类

可以创建任意多个类对象，它们具有相同的属性模式，但可以具有不同的属性值。

2. 可视化组件库 VCL

Delphi的系统库称为Visual Component Library（可视化组件库，简称为VCL）。利用Delphi的可视化组件库，用户只需通过很少的编程就能够创建出复杂的应用程序。

3. Delphi 的组件

Delphi的VCL中提供了大量的组件，每个组件就是一个类，正是由于这些组件的存在，才使得Delphi成为一个方便快捷的应用程序开发工具。

Delphi中的组件按照分类以标签页的形式在Delphi窗体上的组件面板中出现，这些标签页在Delphi窗体上的顺序和编号可能根据Delphi的版本和种类以及安装的选项和附加组件的不同而有所不同。如果已经使用菜单Tool | Environment Options设置了组件面板，那么顺序可能有所不同。

利用组件编程实际上是非常容易的，就是根据需要将按钮、文本编辑框、组合框这样的对象放在窗体中，修改这些组件的属性以及对它们将触发的一些消息进行处理。实际上，窗体也是一个组件，它作为容器可以容纳其他的组件对象。我们在开发Delphi应用程序时，主要是花大量的时间来插入和修改可视化组件对象。

我们可以在Delphi的对象观察器（Object Inspector）或在程序代码中通过对组件对象的属性、方法和事件进行操作来开发应用程序。

属性（Properties）

组件的属性定义了组件对象的行为和特征。通过属性可以得到或者设置该组件的状态、特征等信息。

方法（Methods）

组件的方法是一系列的函数或过程，通过这些方法可对组件进行操作与控制。

事件（Events）

当用户对组件进行某些操作时，如单击它，或系统中某些事情发生时，组件就会产生一个事件。从理论上讲，事件是向窗体发送消息的结果，并且该窗体（或相应组件）可以响应该消息。从技术上讲，尽管事件与消息可能不是一一对应的，但大多数的Delphi事件都是在收到相应的Windows消息后被触发的。Delphi事件的级别比Windows消息的级别要高。

对于Delphi组件，我们要使用它，就要弄清楚它具有一些什么事件，以及各在什么情况下被触发，然后在事件的处理过程中编写处理程序。

1.2 Delphi 7的集成开发环境

运行Delphi 7时的集成开发环境（简称为IDE）如图1.1所示。

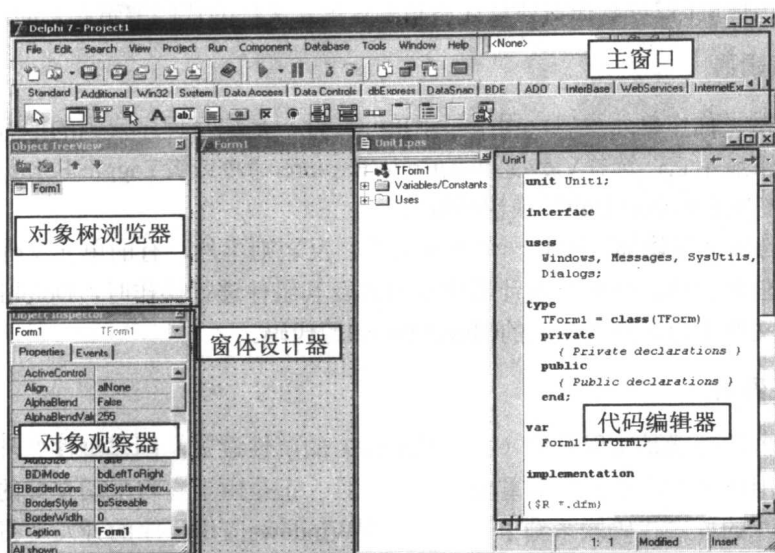


图 1.1 Delphi 7 的 IDE

从图1.1中可以看到Delphi 7的开发环境可分为以下几部分：

1. 主窗口

主窗口位于整个开发环境的顶部，它是开发环境的核心，开发环境中其他几个部分都受主窗口的控制和管理。从上往下依次为：标题栏、菜单栏、快捷工具栏和组件面板。

标题栏在主窗口的顶端，它显示了当前的工程名。如果Delphi 7处于激活状态，标题栏的颜色为蓝色。

Delphi的绝大部分操作命令都可以通过菜单栏上的相应菜单项实现，Delphi的菜单栏在形式上与Windows应用程序菜单栏具有共性。

快捷工具栏上的按钮代表的是菜单命令的快捷形式。按钮上的各种图标很直观地表明了它所实现的功能。快捷按钮都带有提示功能，将鼠标移到任意按钮上不久，就会出现该按钮的简要说明，实际上用Delphi开发的应用程序也具有这种功能。

组件面板是一个双层工具栏，它包含了IDE中安装的所有VCL组件和ActiveX组件。各标签页以及组件在标签页上的顺序和外观可以通过右击它或从主菜单中选择Component | Configure Palette进行调整。

2. 窗体设计器

窗体是将来在应用程序中显示的界面。窗体设计器的初始状态是一个空白的窗体。在程序设计阶段，通过向窗体中添加组件以及更改窗体与窗体中组件的大小、位置等来完成窗体的设计。可以用鼠标调整组件在窗体设计器上的位置和大小，还可以用对象观察器和代码编辑器来控制组件的外观和行为。应用程序的实现正是用事件的形式将窗体及其中的组件的属性与程序代码结合起来的过程。

一般一个窗体对应一个采用Pascal语言编写的单元文件。通过菜单命令View | Toggle

Form| Unit 或快捷键F12可以在窗体设计器和编辑单元文件的代码编辑器之间进行切换。

3. 代码编辑器

代码编辑器又称作单元窗口，用于显示单元文件，是编写程序代码的地方。可以通过以下几种方法来显示它：选择View菜单的Code Explorer命令或者Toggle Form | Unit命令；单击快捷工具栏中的Form | Unit转换快捷键等。

Delphi中的每个窗体都对应着一个单元文件，反之则不然，有的单元文件只包含程序代码而不对应任何窗体。在窗体设计器中对窗体及其组件进行操作时，Delphi的 IDE会在其对应的单元文件中自动加入相应的Object Pascal源代码。

4. 对象观察器

对象观察器的功能是实现对象（尤其是组件）的属性设置，创建事件处理过程并进行管理。它是应用程序设计过程中最重要的一个工具。正是由于它将生动友好的可视化窗体与原本枯燥无味的程序代码联系到了一起，使得Windows应用程序的开发变得如此轻松。我们无需探究乏味的编程语言就可以开发出可视化的Windows应用程序。

对象观察器由三个部分组成：对象列表、属性页和事件页。其中对象列表是一个组合框，包含了当前窗体上的所有组件。属性页中列举了当前被选中的对象（例如组件）的属性。可以在程序设计阶段直接对这些属性进行修改，也可以在程序运行阶段通过执行代码对这些属性进行修改。通常情况下，事件的处理过程为空。可以双击事件右面的组合框来添加事件的处理过程。如果要共用已存在的事件处理过程，可通过下拉式组合框选择已存在的事件处理过程。

5. 对象树浏览器

对象树浏览器位于IDE的左侧，对象树浏览器中以树状图的形式显示在一个Form、Data Module或Frame上的可视组件和非可视组件之间的逻辑关系。

对象树浏览器位于对象观察器的上方，如果IDE中没有显示对象树浏览器，可通过选择菜单View | Object TreeView或按Alt+Shift+F11键来打开它。

6. 工程管理器

选择View菜单的第一项就可以打开工程管理器（Project Manager）。Delphi 7的工程管理器通过工程组进行管理，其中列出了反映各工程之间联系的树状表，使我们可以很直观地找到各工程之间的联系，方便地将相互联系的工程中共同使用的动态链接库文件与可执行文件组织在一起。

1.3 Delphi中的文件及其结构

Delphi的一个工程中包含的最主要文件有：单元文件、窗体文件和工程文件，下面分别介绍这些文件及其结构。

1.3.1 单元文件及其结构

单元文件是带有.pas扩展名的源程序代码。单元文件有三种类型，最常见的一种是与窗体相对应的单元文件，即一个窗体对应一个单元文件；另一种是只存储函数和过程，与窗体没有直接联系的单元文件；还有一种是制作组件的单元文件。

1. 标准的单元文件结构

一个标准的单元文件格式如下：

```
unit Unit1; // 单元头
interface
uses      // 单元列表
//////////
// 接口部分
//////////
implementation
uses      // 单元列表
//////////
// 实现部分
//////////
initialization
//////////
// 初始化部分
//////////
finalization
//////////
// 结束部分
//////////
end.
```

下面具体介绍各部分的功能。

单元头

在单元头部分指定单元的名称。

接口部分

接口部分从保留字interface开始到保留字implementation结束。

在接口部分可以有uses语句，还可以声明常量、数据类型、变量、过程和函数等。如果接口部分有uses语句，则该语句必须紧跟着保留字interface，然后才可以有其他语句。

在接口部分声明的常量、数据类型、变量、过程和函数等，在整个程序中都可以被访问。也就是说，如果在第一个单元中的uses语句中包含了第二个单元，那么在第二个单元的接口部分声明的常量、数据类型、变量、过程和函数等都可以在第一个单元中被访问。

实现部分

实现部分从保留字implementation开始到保留字initialization结束。

在实现部分可以有uses语句。如果实现部分有uses语句，则必须紧跟着保留字implementation，然后才可以有其他语句。

在实现部分也可以声明常量、数据类型、变量、过程和函数等。这些声明通常具有全局隐含的性质，也就是可以在本单元中被访问，但不可以在其他单元中被访问。另外，在实现部分还可以定义过程体和函数体。

在一个单元的interface或者implementation部分都可以定义类，但是类的方法必须在implementation部分实现。

初始化部分

初始化部分从保留字initialization开始到保留字finalization结束。

在初始化部分可以进行一些初始化工作，例如给定义的变量初始化、分配内存空间等。

结束部分

结束部分从保留字finalization开始。

结束部分的工作与初始化部分的工作基本相对应，可以进行一些结束之前的工作，例如释放内存资源等。

只有在单元中出现了初始化部分才可以出现结束部分。两者都是可选项。

2. 单元文件代码分析

一个简单应用程序中的窗体对应的单元文件的代码如下：

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls;

type
  TForm1 = class(TForm)
    Button1: TButton;
    Label1: TLabel;
    procedure Button1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;
```



```

implementation

{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);
begin
    Label1.Caption:='Hello,world!';
end;

end.

```

下面对本单元的主要部分进行简单的分析：

- **unit:** unit标志着下面的代码是一个单元，后面紧跟的是单元文件，其扩展名为.pas。
- **interface:** interface标志着单元接口部分的开始，单元接口部分定义变量、类型、过程等，是其他单元或程序其他部分可以访问的部分。单元接口部分结束于implementation部分的开始。
- **uses:** uses语句告诉程序最终的执行代码中需要哪个函数和过程。Delphi会自动把一些必须的单元包括进来，例如本程序的Windows、Messages和SysUtils等。对于编程人员自己编写的单元，如果程序中使用了该单元的函数或代码，则需要将其加在uses部分中。
- **type:** 这部分用来给编程人员定义自己的类型。此处定义了一个从TForm派生出的TForm1类，它包含一个按钮和一个标签对象，以及一个处理按钮单击事件的过程。
- **var:** 这部分用来定义变量和对象变量。var也用来定义其他implementation部分的变量、过程、函数等。
- **implementation:** 这部分定义所有在interface部分说明过的函数、过程。实际上，implementation部分也可以定义变量、过程、函数等，但只能被本单元局部访问；也可以在这部分增加一个uses语句，来访问其他单元。此处定义了处理单击按钮事件的过程。
- **{SR*.dfm}:** 该语句把单元连接到窗体的.dfm文件。不能删除这条语句，否则会发生编译错误。
- 最后，用一个end加一个英文句号(.)来标志程序的结束。

1.3.2 窗体文件及其结构

窗体文件（DFM文件）用来描述窗体的特性。将窗体文件直接调入Delphi代码编辑器中，窗体文件将转换成文本形式。注意，当窗体文件以文本形式显示时，可以在编辑器弹出菜单中选择View As Form命令返回窗体。

现在分析窗体文件的结构。下面是一个窗体文件的文本形式：

```

Object Form1: TForm1           // 窗体对象
Left = 207

```