

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

DELPHI

《跟我学计算机语言》系列

鹏达工作室 编 著

入门与技巧

武汉大学出版社

《跟我学计算机语言》系列

Delphi 5.0 入门与技巧

鹏达工作室 编著

武汉大学出版社

图书在版编目(CIP)数据

Delphi 5.0 入门与技巧/鹏达工作室编著. —武汉: 武汉大学出版社,
2000. 12

(跟我学计算机语言系列)

ISBN 7-307-03078-0

I. D… II. 鹏… III. DELPHI 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2000)第 43672 号

责任编辑: 毕卫东 责任校对: 王 建 版式设计: 支 笛

出版: 武汉大学出版社 (430072 武昌 珞珈山)

(电子邮件: wdp4@whu.edu.cn 网址: www.wdp.whu.edu.cn)

发行: 新华书店湖北发行所

印刷: 核工业中南三〇九印刷厂

开本: 787×1092 1/16 印张: 13 字数: 258 千字

版次: 2000 年 12 月第 1 版 2000 年 12 月第 1 次印刷

ISBN 7-307-03078-0/TP·90 定价: 18.00 元

版权所有, 不得翻印; 凡购买我社的图书, 如有缺页、倒页、脱页等质量问题者, 请与当地图书销售部门联系调换。

内 容 简 介

《跟我学语言》系列丛书 Delphi 入门与技巧共分为 11 章,详细讲解了 Delphi 的特性、Delphi 的可视化部件和非可视化部件、工程管理与设计工具及数据库技术。对于 Delphi 的设计语言——Object Pascal,运用典型的语句及简单的例程加以说明,避免生硬的概念和繁琐的语法。本书涉及 Delphi 应用与编程的方方面面。此外还备有大量的附录,内容经典,资料详实,以供读者查询。

AB-110

前 言

目前,各种语言编程资料和教程书众多,但大部分由于内容繁杂,概念抽象空洞,专业词汇层出不穷,使一般想学编程的初学者敬而远之,遇到问题时无所适从,而对于一般的人门者,往往没有提供一个行之有效的快速提高方法和途径,使之能尽快提升自己的编程水平。在此,我们编写了《跟我学语言》系列丛书,并且精心制作了《编程经典》系列和《用多媒体学》系列教学光盘(另售)。在这套丛书中我们本着实用、即时、快学的原则,以精辟详实的理论讲解和丰富经典的例子剖析,使您迅速进入编程之路的捷径。

Delphi 入门与技巧这本书共分为 11 章,详细讲解了 Delphi 的特性、Delphi 的可视化部件和非可视化部件、工程管理与设计工具及数据库技术。对于 Delphi 的设计语言——Object Pascal,运用典型的语句及简单的例程加以说明,避免生硬的概念和繁琐的语法。本书涉及 Delphi 应用与编程的方方面面。此外还备有大量的附录,内容经典详实,以供读者查询。

该系列丛书适合入门者和面临应用的初、中级用户,也适合大学、中专计算机专业和非计算机专业自学使用。本书由武汉鹏达计算机系统开发有限公司“鹏达工作室”刘昌永、陈进征、钱劲松等编著,在此特别感谢相关工作人员和技术编辑。

由于时间仓促,错误和不当之处敬请一并指出!

鹏达工作室
2000 年 10 月

目 录

第一章 Delphi 5 简介	1
1.1 Delphi 的特性	1
1.2 Delphi 5 的增强功能	2
1.2.1 Delphi 5 的用户接口增强功能	2
1.2.2 Delphi 5 先进的工程管理	3
1.2.3 Delphi 5 使用先进的调试技术,加速软件的发布	5
1.2.4 Delphi 5 最新编程语言的特点简化了开发过程	5
1.2.5 可开发 Windows 和 BackOffice 的应用程序	5
1.2.6 成熟的数据库访问和决策支持应用程序的开发	6
1.2.7 可方便地开发分布式应用程序和 Internet 应用程序	6
第二章 Delphi 5 快速入门	8
2.1 安装 Delphi 5	9
2.2 进入 Delphi 5 环境	10
2.3 Delphi 5 的界面组成	11
2.4 设计简单的用户界面	14
2.4.1 开始设计	14
2.4.2 选取部件加入到窗体中	14
2.4.3 部件的调整与对齐	15
2.4.4 锁定部件	17
2.4.5 保存所做的工作	17
2.4.6 运行工程	17
2.4.7 改变对象的属性	18
2.4.8 编写事件处理过程	20
2.5 使用联机帮助 Help	21
第三章 Delphi 5 的常用可视化部件	22
3.1 常用的与文本相关的部件	22
3.1.1 Label 部件	22
3.1.2 Edit、MaskEdit 和 Memo 部件	23

3.1.3	ListBox 和 ComboBox 部件	25
3.2	按钮和检查框部件	27
3.2.1	Button 和 BitBtn 部件	27
3.2.2	SpeedButton 部件	28
3.2.3	CheckBox 与 RadioButton 部件	28
3.3	分组、分界部件	29
3.3.1	GroupBox、RadioGroup 及 Panel 部件	29
3.3.2	NoteBook、TabSet 及 TabbedNoteBook 部件	29
3.3.3	分界部件	30
3.4	图形、图像部件	31
3.4.1	Image 部件	31
3.4.2	Shape 部件	31
3.4.3	PaintBox 部件	32
3.5	关系图、文件列表部件	33
3.5.1	Outline 部件	33
3.5.2	目录访问部件	33
3.6	滚动部件	35
3.6.1	ScrollBar 部件	35
3.6.2	ScrollBar(滚动框) 部件	36
3.6.3	几个进度显示部件	36
3.7	网格、表格部件	37
3.8	多媒体 (MultiMedia) 和 OLE 部件	39
 第四章 使用非可视化部件		41
4.1	菜单	41
4.1.1	主菜单 (MainMenu) 的设计	42
4.1.2	为菜单添加事件	43
4.1.3	创建嵌套菜单	44
4.1.4	给菜单加提示 (Hint)	45
4.1.5	菜单的其他属性	45
4.1.6	弹出式菜单 (Popup Menu) 与主菜单的区别	45
4.2	计时器部件 Timer	46
4.2.1	Timer 的属性	46
4.2.2	使用 Timer 对程序进行控制	46
4.3	使用公用对话框部件	48

4.3.1 调用标准对话框	48
4.3.2 OpenFileDialog 和 SaveDialog 部件	48
4.3.3 ColorDialog 和 FontDialog 部件	49
4.3.4 PrintDialog 和 PrinterSetupDialog 部件	50
4.3.5 FindDialog 和 ReplaceDialog 部件	51
第五章 Delphi 的工程管理和设计工具	52
5.1 创建多窗体工程项目	52
5.1.1 创建一个含有 About 框的例程	52
5.1.2 指定自动创建窗体	55
5.2 使用工程管理器 Project Manager	57
5.3 工程文件的组成	58
5.4 使用窗体样板和对话框专家	58
5.4.1 使用窗体样板创建窗体	58
5.4.2 存储自己的窗体样板	59
5.4.3 使用对话框专家 Dialog Wizard	60
5.5 使用工程样板和应用专家	60
5.5.1 使用工程样板	60
5.5.2 使用应用专家 Application Wizard	61
第六章 Object Pascal 语言初步	63
6.1 编写 Object Pascal 程序代码	63
6.1.1 编写赋值语句	63
6.1.2 标识符的说明与使用	64
6.1.3 变量	64
6.1.4 预定义类型	65
6.1.5 常量	66
6.2 过程与函数	66
6.2.1 一个调用 Delphi 方法的简单例程	67
6.2.2 调用 Delphi 的含参过程	68
6.3 跳转语句	69
6.3.1 if 语句	69
6.3.2 case 语句	70
6.4 循环语句	71
6.4.1 repeat 语句	71

6.4.2	while 语句	72
6.4.3	for 语句	72
6.5	程序模块	73
6.6	关于作用范围	73
6.6.1	标识符的作用范围	73
6.6.2	访问其他程序模块中的说明	74
6.6.3	按照作用范围说明标识符	75
6.7	编写一个过程或函数	76
6.7.1	一个自行编写的函数例程	76
6.7.2	过程和函数的标题	77
6.7.3	函数和过程中的类型说明	77
6.7.4	过程和函数的语句部分	78
6.7.5	函数的递归调用	78
6.7.6	过程和函数的参数	79
6.8	定义新的数据类型	80
6.8.1	枚举类型	81
6.8.2	子界类型	81
6.8.3	数组类型	81
6.8.4	字符串类型	82
6.8.5	集合类型	83
6.8.6	记录类型	83
6.9	Object Pascal 的库单元 Unit	84
6.9.1	Object Pascal 程序库单元的结构	85
6.9.2	程序库单元的接口部分	85
6.9.3	程序库单元的实现部分	86
6.9.4	程序库单元的初始化部分	86
6.9.5	使用 Delphi 的可视化部件及其库单元	86
6.9.6	建立与窗体无关的新库单元	87
6.9.7	将库单元加入工程	87
第七章	用 Delphi 的对象进行编程	88
7.1	什么是对象	88
7.1.1	对象的封装性	88
7.1.2	对象的继承性	88
7.1.3	对象的多态性	88

7.1.4 通过 Delphi 实例了解对象	89
7.2 从一个对象中继承数据和方法	92
7.3 对象的范围	93
7.3.1 什么是对象的范围	93
7.3.2 重载一个方法	95
7.4 对象公有域和私有域的说明	95
7.5 访问对象的域和方法	95
7.6 对象变量的赋值	97
7.7 建立非可视化对象	99
7.7.1 说明一个非可视化对象	99
7.7.2 用 Create 方法建立对象实例	99
7.7.3 撤销对象	100
7.8 数据库应用程序	100
第八章 数据集概述	104
8.1 打开和关闭数据集	104
8.2 浏览记录	105
8.3 书签	106
8.4 搜索特定的记录	107
8.5 过滤	108
8.6 修改数据	109
8.7 事件	109
第九章 访问表格	111
9.1 使用 TTable 构件的一般步骤	111
9.1.1 设置 TTable 构件	111
9.1.2 指定要访问的数据库	111
9.1.3 指定要访问的表格	114
9.1.4 指定表格的类型	114
9.1.5 打开和关闭一个表格	114
9.1.6 控制对表格的读写	115
9.2 在表格中搜索记录	115
9.2.1 基于索引中的字段搜索记录	115
9.2.2 Goto 系列	116
9.2.3 Find 系列	116

9.2.4	KeyExclusive 属性和 KeyFieldCount 属性	117
9.2.5	基于副索引进行搜索	117
9.3	对记录排序	117
9.3.1	检索已定义的索引名	118
9.3.2	指定使用哪个副索引	118
9.3.3	检查索引中的字段	118
9.4	选择部分记录	119
9.4.1	过滤和范围的区别	119
9.4.2	设置范围的起点	119
9.4.3	设置范围的终点	120
9.4.4	同时设置起点和终点	120
9.4.5	近似匹配	121
9.4.6	应用和取消范围	121
9.4.7	修改范围的起点和终点	121
9.5	对表格整体的操作	122
9.5.1	删除表格中的所有记录	122
9.5.2	删除一个表格	122
9.5.3	给表格换名	123
9.5.4	创建表格	123
9.5.5	两个表格之间保持同步	125
9.6	Master/Detail 关系	125
9.7	嵌套表	128
9.8	从另一个表格中引入数据	129
9.9	使用 TBatchMove	129
9.9.1	使用 TBatchMove 构件的一般步骤	129
9.9.2	指定操作方式	130
9.9.3	映射字段类型	131
9.9.4	执行批量移动操作	131
9.9.5	处理错误	131
第十章	查询数据库	133
10.1	如何查询数据库	133
10.1.1	查询桌面数据库	133
10.1.2	查询远程数据库	133
10.1.3	可以查询哪些数据库	134

10.2 使用 TQuery 构件的一般步骤	134
10.3 指定要执行的 SQL 语句	135
10.3.1 概述	135
10.3.2 在设计期指定 SQL 语句	135
10.3.3 在运行期指定 SQL 语句	136
10.4 参数	137
10.4.1 在设计期提供参数	137
10.4.2 在运行期提供参数	138
10.4.3 从另一个数据集获得参数	139
10.5 执行查询	139
10.6 异构查询	140
10.7 查询结果	140
第十一章 如何使用其他数据控件	142
11.1 Delphi 5 中有哪些数据控件	142
11.2 数据控件的基本用法	143
11.2.1 指定一个数据源	143
11.2.2 编辑和更新数据	143
11.2.3 禁止和允许数据刷新	143
11.2.4 手动刷新数据	144
11.3 显示单个字段的数据控件	144
11.3.1 把数据作为标签显示	144
11.3.2 显示和编辑数据	145
11.3.3 显示和编辑多行文本	145
11.3.4 以 RTF 格式显示文本	146
11.3.5 显示和编辑图像	146
11.4 用列表框和组合框显示和编辑数据	146
11.4.1 TDBListBox	147
11.4.2 TDBComboBox	147
11.4.3 显示另一个数据集中的数据	147
11.5 用复选框处理布尔类型的字段	148
11.6 用单选分组框限制字段的值	148
11.7 使用 TDBGrid	149
11.7.1 动态的列对象	149
11.7.2 永久的列对象	149

11.7.3	在设计期设置列对象的属性	151
11.7.4	显示 ADT 和数组字段的值	151
11.7.5	设置栅格的选项	152
11.7.6	在运行期响应用户的动作	152
11.7.7	特殊的数据库栅格	153
11.8	导航器	154
11.9	数据源	155
附录一	编译错误信息	158
附录二	运行错误信息	169
附录三	相关类库	185
附录四	快速索引	187

第一章 Delphi 5 简介

1.1 Delphi 的特性

Delphi 5 是 INPRISE 公司最新推出的一件优秀软件产品,适用于初学者和专业程序员进行软件开发。使用它可以在很短的时间内创建自包含式的、用户界面友好的 Windows 应用程序。而且所创建程序的窗口是在屏幕上“绘制”出来的,始终可以看到程序最终的执行效果,所以一经推出,就风靡全球。

Delphi 是一种先进的开发工具,它具有以下特点:

- (1) 将真正的编译器与真正的视觉化开发环境结合在一起。
- (2) 拥有双向代码生成器。
- (3) 完整的、一步到位的 ActiveX 支持。
- (4) 拥有高性能的数据库访问结构。
- (5) 支持跨平台的 Internet。

Delphi 将可视化技术和面向对象技术完美地结合在一起,突破了应用程序开发越来越复杂的障碍,让开发人员将注意力较多地集中在业务需求的开发上,而较少注意底层 API 调用、工程管理配置、代码导航等。

Delphi 的新版本 Delphi 5 有哪些新特点?

在 Delphi 的新版本 Delphi 5 中,INPRISE 公司仍让其使用者保持在技术的最前沿。Delphi 5 增加了众多的新特性来满足软件开发过程中各个阶段的需求:

- (1) 最新改进的用户接口更加友好、易用,详见下节 Delphi 5 的用户接口增强功能部分。
- (2) 先进的工程管理和代码浏览功能,大大减少了管理复杂工程及快速定位代码的麻烦。详见下节 Delphi 5 的先进的工程管理部分。
- (3) 功能强大的全新调试工具可以更快地发现更多的 bug。
- (4) 新增程序设计语言特性,包括 Action Lists,这是用户接口开发方面的巨大飞跃。
- (5) 全面支持 Windows 98、Windows NT 和 Oracle 8。
- (6) 能完整地一步到位地开发 COM(组件对象模型)和 CORBA 组件,一个组件同时支持 COM 和 CORBA 两种标准。

(7)与 Java 有良好的互操作性。

1.2 Delphi 5 的增强功能

1.2.1 Delphi 5 的用户接口增强功能

Delphi 5 的用户接口增强了哪些功能?

Delphi 的用户接口设计得非常友好,适合于任何开发人员快速、高效地进行应用程序开发。几乎每个功能都是为开发人员定制的,无论是新手还是老手都会觉得使用起来得心应手。Delphi 5 完全支持 Docking Windows 和 Forms,这意味着代码窗口、窗体和调试窗口可以很方便地被重新分组、排列,这样在项目开发的各个阶段开发人员都可以选择更适合的桌面,如图 1-1 所示。

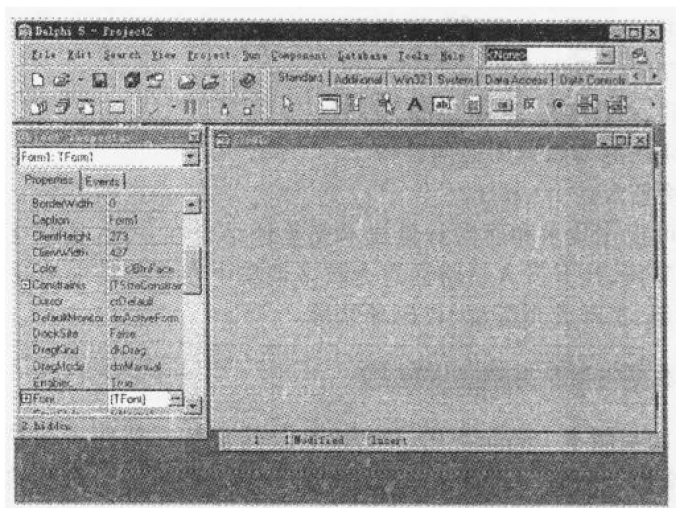


图 1-1 Delphi 的界面

Delphi 5 是 Tabbed Docking Windows 技术的先导。开发者可以将很多窗口归并到一个极大化的选项卡式的窗口中,再也无需在多个窗口间进行切换。如果安装了 Intellimouse(智能鼠标),Delphi 可以使用精确的代码条滚动。开发人员可以使用此环境设计出一流的用户接口模块。

1.2.2 Delphi 5 先进的工程管理

Delphi 5 的工程管理有哪些优点?

应用程序由许多组件元素组成:源代码、窗体、队列、DLLs、ActiveX 组件、帮助文件……,并且组件元素还在不断地增加。追踪每一个组件的状态(依赖关系、是否选用、版本号等)是非常必要的,这可提高开发的效率。

Delphi 5 多任务的 Advanced Project Manager(高级项目管理器)可以从同一源代码中,在同一时间内重建组件、DLLs 和可执行文件,而版本号是绝不会重复的。

什么是 AppBrowser 技术?

AppBrowser 这一革新技术是 INPRISE 研究开发实验室的最新研究成果。App-Browser 用先进的后台分割来理解各部分代码(如方法调用、定义和执行)的内在关系。然后只需通过单击鼠标,使用幕后的智能超链接(hyperlinks)便可找到你所需的源代码。在代码间跳跃就像在 Internet 的 Web 文档上冲浪一样简单,如图 1-2 所示。

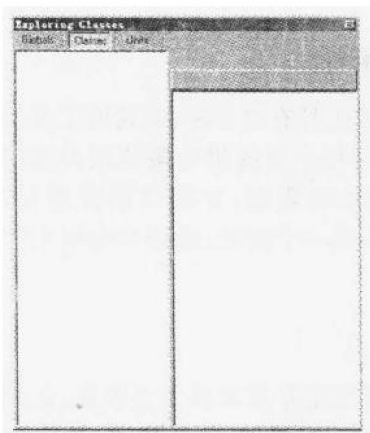


图 1-2 Browser 窗口

什么是 CodeExplorer?

CodeExplorer 为每个工程建立一个与其代码实时保持一致的“Table of Contents (内容表)”,给每个工程提供一个直观的层次结构,使开发人员从高层面纵观整个工程,能方便地跳转到他所关心的任何具体代码位置,如图 1-3 所示。

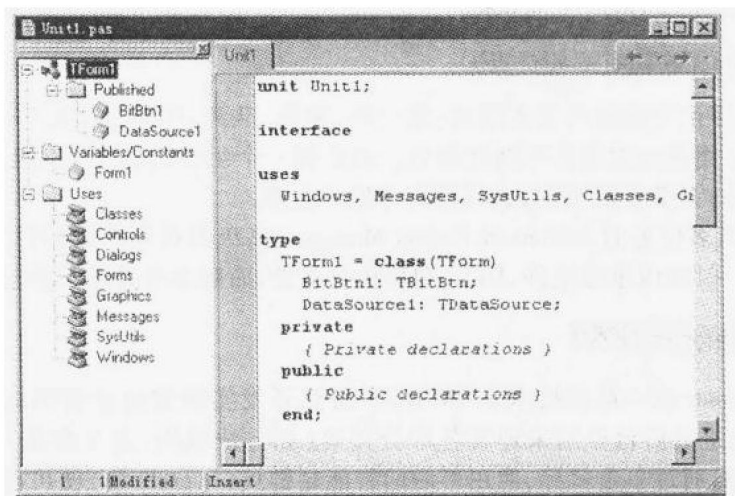


图 1-3 CodeExplorer 窗口

什么是 ClassCompletion Wizard?

Delphi 的强大代码生成工具有口皆碑,而其他工具均需使用“黑盒子”代码生成器来自动编码,因而只能向程序员提供有限范围内的编辑、定制最终代码的功能。Delphi 的双向工具却没有这种限制,它提供给程序员对代码的完全控制。新的 ClassCompletion 向导自动生成一个新类,使用 Delphi 的双向技术可实时地保持向导和源代码之间的同步。

什么是 ActionLists?

ActionLists 可能是继视觉化开发工具诞生以来,在用户接口开发方面最有意义的效率提升。它使多种用户接口组件(例如热键、工具条按钮、菜单项目和弹出式菜单等)可以在不同的应用对象中建立同一个命令(或行为),开发人员可以有效地管理大量用户接口与应用逻辑之间的连接。通常改变一个简单的用户命令需要从所有应用程序中查找出与建立这个命令相关的用户接口组件,而 ActionLists 具有管理这些复杂关系的功能,它使得开发人员能将注意力集中在业务逻辑的开发上。相反,没有 ActionLists,开发者必须跟踪每一个命令与应用对象间的连接。很多时候用户接口组件(如粘帖命令)在处理失效或是重新生效是基于一个应用对象的特殊状态,这就导致用户接口的设计很复杂。有了 ActionLists,开发人员只用一个 Action Item 来管理用户接口与应用逻辑间的一个连接。通过维护这个简单的 Action,开发人员才有可能用真正的面向对象的方式来管理应用逻辑与源代码间的互操作。Delphi 5 提