



Delphi程序设计

Delphi CHENGXU SHEJI

(第2版)

■ 王行言 主 编



国家开放大学
THE OPEN UNIVERSITY OF CHINA



国家开放大学新型产业工人
培养和发展助力计划

Delphi 程序设计

(第2版)

王行言 主 编

中央广播电视大学出版社·北京

图书在版编目 (CIP) 数据

Delphi 程序设计/王行言主编. —2 版. —北京: 中央广播电视大学出版社, 2016. 1

国家开放大学新型产业工人培养和发展助力计划

ISBN 978 - 7 - 304 - 07681 - 8

I. ①D… II. ①王… III. ①软件工具—程序设计—开放大学—教材 IV. ①TP311. 56

中国版本图书馆 CIP 数据核字 (2015) 第 318062 号

版权所有, 翻印必究。

Delphi 程序设计 (第 2 版)

Delphi CHENGXU SHEJI

王行言 主 编

出版·发行: 中央广播电视大学出版社

电话: 营销中心 010 - 66490011

总编室 010 - 68182524

网址: <http://www.crtvup.com.cn>

地址: 北京市海淀区西四环中路 45 号

邮编: 100039

经销: 新华书店北京发行所

策划编辑: 邹伯夏

版式设计: 赵 洋

责任编辑: 王 可

责任校对: 宋亦芳

责任印制: 赵连生

印刷: 北京明月印务有限责任公司

印数: 0001~1000

版本: 2016 年 1 月第 2 版

2016 年 1 月第 1 次印刷

开本: 787mm × 1092mm 1/16

印张: 19.5 字数: 462 千字

书号: ISBN 978 - 7 - 304 - 07681 - 8

定价: 29.00 元

(如有缺页或倒装, 本社负责退换)

PREFACE 前言

Delphi 是一个基于面向对象和组件技术的可视化的开发环境，它使用对象 Pascal 作为基本的编程语言，可以用来开发各种基于图形界面的应用程序。

数据库应用程序的开发一直是 Delphi 的强项，Delphi 支持从 Access 到 Oracle 的各种数据库类型。

Delphi 因其完全可视化的开发环境、丰富的组件支持、与 Windows 系统的高度融合，以及速度极快的编译器，一直为广大专业的开发人员所推崇。Delphi 是一个优秀的、专业级的开发工具。

Delphi 一直紧紧抓住软件技术发展的大趋势。从过去的组件技术到基于 Web 的应用开发，再到移动应用的开发，Delphi 都有非常良好的技术支持和先进的解决方案。

本书是一本有关 Delphi 程序设计的基础性教材。本书由浅入深，比较详细地介绍了面向对象与组件的基本概念，以及对象 Pascal 的基本语法。本书依次介绍了 Delphi 的开发环境与开发方法，介绍了常用组件的使用，还重点介绍了用户界面设计，以及在数据库和多媒体等方面的应用开发。

本书既讲述一些重要的概念和原理，同时也讲述一些实用性的技术。本书通过大量的例题，引导读者从掌握开发工具的使用逐步深入到如何利用 Delphi 的功能进行应用程序的开发。

全书共分为 7 章，主要内容如下：

第 1 章介绍 Delphi 的基础知识，包括 Delphi 的开发环境与基本开发过程。

第 2 章全面介绍常用基本组件的使用，并使读者进一步理解组件的属性、方法和事件的基本概念，以及 Delphi 的编程思路。

第 3 章介绍常用 Delphi 进阶组件的使用，包括列表框、状态条、时钟组件和对话框等。

第 4 章介绍用户界面设计的基本原则和相关技术，如菜单、工具栏、多页面，以及多窗体的实现等。

第 5 章介绍 Delphi 主要的多媒体组件及应用。

第 6 章介绍 Delphi 的数据库开发环境，并详细介绍了 Delphi 数据库组件及应用。

第 7 章通过一个综合实例，介绍如何利用 Delphi 的多项开发技术，开发一个小型数据库应用系统。

本书每章开头都有“内容提要”和“学习目标”，每章的结束部分都有“本章小结”和“本章实验”便于读者更好地掌握每一章的内容，也便于教师组织教学。各章还附有习题，其中包括思考题和上机题。

本书第1章、第2章、第4章由清华大学王行言教授和国家开放大学王欣副教授编写；第3章和第5章由王欣编写，第6章由北京家友科技有限公司高级工程师部凡编写，第7章由王行言和部凡编写。全书由王行言统一定稿。

由于时间仓促，书中的不当之处在所难免，敬请广大读者批评指正。

编者
2015年11月

CONTENTS 目录

第1章 Delphi 基础知识	1
1.1 初识 Delphi	1
1.2 Delphi 的集成开发环境	2
1.3 Delphi 应用程序的开发过程	10
1.4 例子：第一个 Delphi 程序	10
1.5 代码编辑与代码调试	18
1.6 Object Pascal 的语法基础——变量和运算	27
1.7 Object Pascal 的程序结构	32
1.8 项目管理	40
1.9 在集成环境中获得联机帮助	47
本章小结	51
第2章 Delphi 基本组件及应用	54
2.1 组件的概念	54
2.2 基本文本输入/输出组件	57
2.3 按钮类组件	64
2.4 Object Pascal 语法——控制语句	75
本章小结	82
第3章 Delphi 进阶组件及应用	84
3.1 其他常用可视组件	84
3.2 时钟组件	100
3.3 标准对话框组件	104
3.4 语法——构造数据类型	112
本章小结	122
第4章 用户界面设计	125
4.1 用户界面设计的基本原则	125
4.2 菜单与工具栏设计	126
4.3 多页面设计	145

4.4 窗体设计	152
4.5 编写过程与函数	160
本章小结	170
第5章 多媒体组件及应用	174
5.1 Windows 中的常见多媒体文件	174
5.2 Delphi 中的常用多媒体组件	176
5.3 图像媒体 (Image) 组件	176
5.4 动画 (Animate) 组件	185
5.5 媒体播放 (MediaPlayer) 组件	192
本章小结	200
第6章 数据库组件及应用	202
6.1 关系数据库管理系统 Microsoft Office Access	202
6.2 Delphi 数据库组件	218
6.3 数据集组件及应用技术	238
6.4 数据集的访问与字段对象	249
6.5 数据集组件——Table	263
6.6 数据集组件——Query	267
6.7 数据库组件应用实例	274
本章小结	284
第7章 综合实例	287
7.1 系统分析与设计	287
7.2 系统实现	288
本章小结	305
参考文献	306

第 1 章 Delphi 基础知识

本章简要介绍 Delphi 软件的概貌，介绍集成开发环境的界面组成与使用。本章通过一个例子扼要介绍 Delphi 应用程序的开发过程，以及与此相关的概念、技术和方法，使读者对 Delphi 有一个感性认识。另外，还简要介绍 Object Pascal 的程序结构、变量和计算，为后面章节的学习打下基础。

有关 Object Pascal 语言的详细内容，可以阅读 Delphi 的联机帮助信息。

1. 理解什么是 Delphi，它主要适合哪些方面的应用。
2. 了解 Delphi 集成开发环境的界面组成、功能及操作方法。
3. 理解 Delphi 应用开发的基本过程，以及每一步骤的任务和实现方法。
4. 了解代码调试的基本方法。
5. 掌握语言基础知识（基本数据类型、运算符、赋值语句、复合语句、注释语句等）。
6. 理解变量的作用域和生命周期，掌握局部变量、单元全局变量的定义。
7. 理解 Delphi 的程序结构，理解单元的结构与组成，理解项目的基本概念，以及项目中文件的构成与含义。
8. 初步掌握在 Delphi 集成环境中获得联机帮助的几种主要途径。

1.1 初识 Delphi

什么是 Delphi？它有哪些特点？它能适用于哪些应用类型的开发？这是读者首先要弄明白的。

Delphi 是面向对象的、可视化的集成开发环境；它使用 Object Pascal 作为核心的编程语言；它可用来开发各种具有图形用户界面的应用程序和系统软件，从桌面应用到多层的 Web 应用，从多媒体到基于数据库的应用系统。可以说，Delphi 是一个专业级的、通用的开发



工具。

Delphi 由 Borland 公司初创，于 1995 年面世，并迅速成为主流开发平台。2009 年后，Delphi 工具改由 Embarcadero 进行升级，称为 RAD Studio XE 系列平台，并成为支持快速构建用于 Windows、Mac OS X、iOS 和 Android 移动开发的新型开发平台。本书基于 Delphi XE8 进行讲解。开发人员可以在 Embarcadero 的官方网站上下载一个 Embarcadero RAD Studio XE8（链接为 http://altd.embarcadero.com/download/radstudio/xe8/delphi_xe8_esd.exe），执行安装文件。Delphi XE8 的安装语言目前只支持英语、法语、德语、日语，因此，一般选择应用 English 选项，然后按照界面要求安装。安装时要注意两项：一是安装计算机必须连接互联网，以便下载安装所需文件；二是系统安装盘中至少保留 20 GB 的空闲空间。

Delphi 所提供的集成开发环境（Integrated Development Environment, IDE）可以提供从系统设计、编码到程序运行、调试所需要的全部工具，并为这些工具提供了很好的连接，为软件开发人员提供了一个功能齐全、便捷、高效的开发环境。Delphi 的集成开发环境由项目管理器、对象查看器、代码编辑器、图像编辑器，以及数据库工具等一系列工具组成，这些工具分工协作，支持开发人员在一个集成环境中完成应用开发的全过程。

Delphi 提供了一个丰富的可视组件库（Visual Component Library, VCL）。VCL 是由一些被称为“组件”（Components）的预制类组成的。VCL 中的组件是用 Delphi 的 Object Pascal 语言编写，并存储于系统库中的。用户所使用的任何一个组件都会成为可执行程序的组成部分。

在 Delphi 中会经常提到“组件”一词。组件是在设计阶段便可对其操作的可视化对象，所有组件类都是从 TComponent 类继承下来的，正是这些组件支持了可视化编程模式。用户界面设计、组件对象的建立、对象属性的初始化等工作都可以用“所见即所得”的方式实现，可视化编程模式大大简化了应用程序的开发过程，降低了编码的难度和工作量。

Delphi 还提供了一套快速应用开发（Rapid Application Development, RAD）模板，包括窗体、菜单和编程向导等。通过这些工具和 Delphi 的本地代码编译器，程序员可以快速创建和测试原型，完成应用程序的开发。

总之，Delphi 所具有的强大功能和良好的开发环境，使它成为众多软件开发人员的首选工具。

1.2 Delphi 的集成开发环境

1.2.1 对象与事件

Delphi 是基于面向对象技术的。在 Delphi 中，“对象”是这样定义的：一个对象是由属性（Properties）、方法（Methods）和事件（Events）所组成的。属性表示对象中的数据，方法是对象能够执行的动作，事件是能够引起对象反应的条件。

基于图形用户界面的应用程序都是“事件驱动型”的，这是由程序的交互性和实时性所决定的。事件驱动型程序的运行并不是按照事先安排好的顺序进行的，程序的执行流程完全依照用户的实际操作和系统的运行情况而定。在程序执行过程中，程序在不断响应用户的指令和各种事件的发生。当程序启动后，如果用户不进行任何操作，则应用程序可以一直处于待命状态；而当应用程序完成用户的一次操作后，程序又会等待用户的下一道命令。

这样的程序采用面向对象方法最为合适，因为事件驱动机制就是模拟现实世界中对象之间的信息传递与相互作用的。Delphi 为大多数对象定义了各自能够响应的事件，即什么样的情况发生会引起一个对象做出反应。不同的对象所能响应的事件是不尽相同的，每个对象都有自己的“事件清单”。

例如，用户界面上的按钮对象可以响应 OnClick 事件。当用户单击按钮时，OnClick 事件发生，按钮就会做出反应，按钮的 OnClick 事件处理程序就会被触发。

又如，行编辑框对象可以响应“数据修改”事件。当用户改动编辑框中的数据时，“数据修改事件”发生，行编辑框就会做出反应，行编辑框的“数据修改事件处理程序”就会被触发。

读者会问，这些事件处理程序究竟完成什么功能呢？这就要看开发人员想让这些对象对事件做出什么反应。例如，界面上的一个按钮是“退出”按钮，开发人员设计这个按钮的功能是，当用户单击这个按钮后，程序结束运行，所以程序员就要编写这个按钮的 OnClick 事件处理程序，并在代码中调用退出系统的有关过程，结束程序的运行。

事件处理程序是需要编写的，没有编写的事件处理程序是不存在的。即使事件出现了，对象也不能做出任何反应。如果一个对象被定义为可对十个事件做出反应，但设计人员根据程序需要，只让该对象对其中的一个事件做出反应，那么程序员就只需编写这个事件的处理程序。

在 Delphi 应用开发中，程序员很主要的一项工作就是根据应用系统的功能设计，决定哪个对象需要对哪个事件做出反应，以及做出什么反应，然后着手编写事件处理程序。

1.2.2 Delphi 的界面组成与使用

当单击图标，启动 Delphi 后，就会进入如图 1-1 所示的操作界面。

Delphi 的操作界面由五个相对独立的窗口组成：主窗口、对象查看器（Object Inspector）、对象结构视图（Structure）、项目管理器（Project Manager）、工具面板（Tool Palette）。

1. 主窗口

如图 1-1 所示窗口的最上方为 Delphi 的主窗口，由菜单栏、工具栏和编辑器组成。主窗口的关闭就意味着退出 Delphi。

在菜单栏中，既包含常规的菜单项，如 File、Edit、Search、View 等，也包含具有 Delphi 特征的菜单项，如 Project（项目）、Component（组件）等。有关菜单栏中各菜单项的功能，我们将结合各章内容陆续进行介绍。

工具栏的默认布局分为左、右两个部分。左侧是快捷按钮区，右侧是 Delphi 界面中最具特色的可视组件面板（Component Palette）。

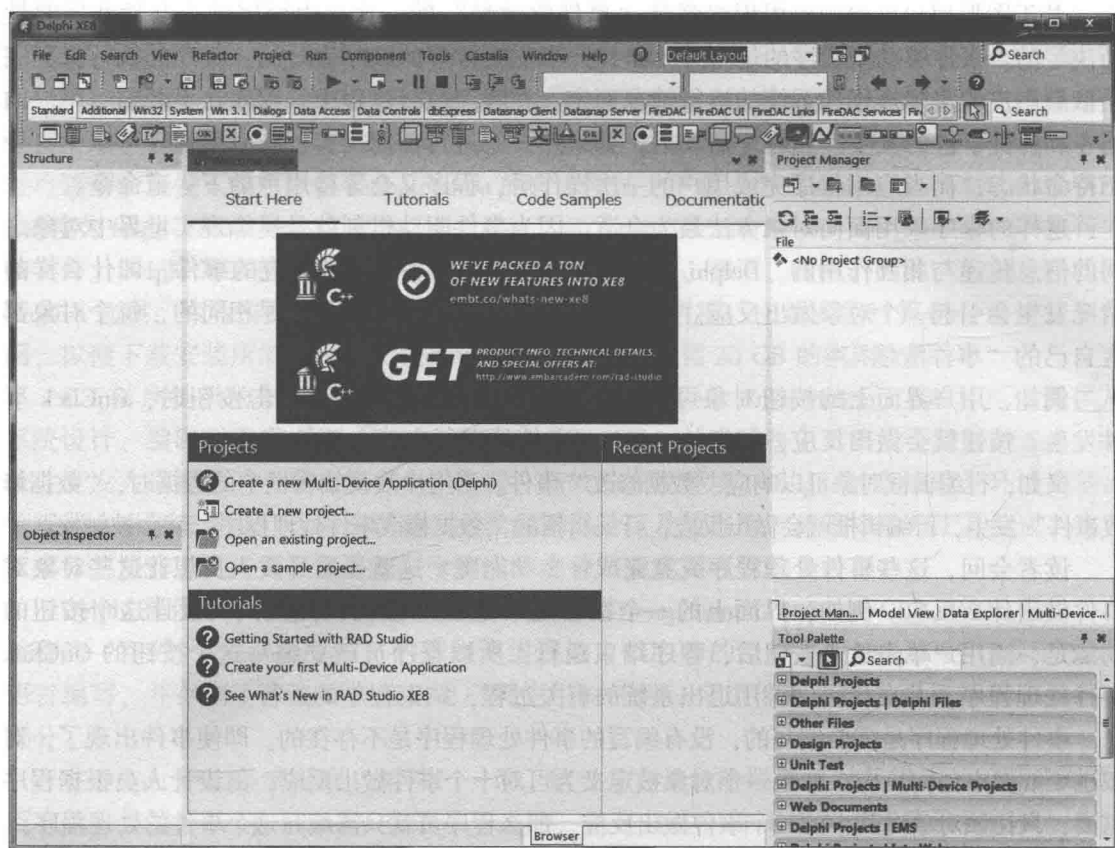


图 1-1 Delphi 的操作界面

快捷按钮区如图 1-2 所示。快捷按钮区的 18 个快捷按钮分别代表了开发过程中时常用到的相应的菜单命令。这些按钮好似就放在手边，为开发人员提供了便捷的操作方式。

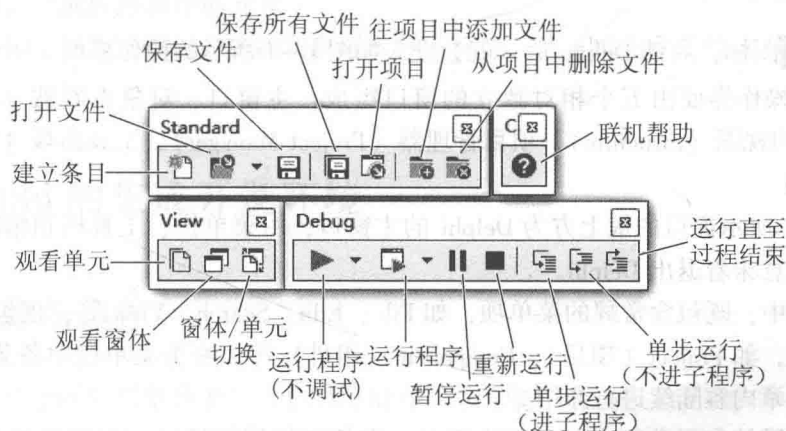


图 1-2 快捷按钮区

在工具栏第一排的8个按钮中,除联机帮助按钮以外,其他7个按钮都是有关项目和文件的操作;在第二排按钮中,有关窗体和单元的3个按钮用于窗体和单元窗口的激活等,其他几个按钮则用于程序的运行与调试。利用菜单项 View | ToolBars 可以定制工具栏。

在 Delphi 中,什么是项目?什么是窗体?什么是单元?文件是不是一般意义上的文件?这些问题将在下文一一介绍。

可视组件面板是在开发环境中摆放可视组件的地方。如图 1-3 所示,系统将可视组件按功能分成若干类,分别放在组件板不同的页面上。从页标签上可以看出每页组件的类别。例如,Standard 是标准组件页,该页包含了读者在 Windows 界面上经常用到的那些组件,如普通按钮、单选按钮、复选按钮、列表框、编辑组件和菜单等。又如,Data Access 和 Data Controls 两个页面上放置的是有关数据库应用开发方面的组件。



图 1-3 可视组件面板

单击页标签便可显示指定页的组件,而将鼠标定位在一个组件图标上,Delphi 会显示该组件的名称。因为页面较多,图 1-3 中并没有显示出全部的页面。通过单击组件板右上角的左右箭头,便可浏览全部的页面。利用鼠标拖动页标签,也可以改变组件页的排列顺序。

准确地说,组件板上的组件是类的概念,是可视化的组件类,而不是具体的组件对象。

Delphi 采用点放式设计方式,只要简单地单击选中组件板上的一个组件图标,就选取了某个组件类,然后在窗体上单击鼠标或拖动鼠标,Delphi 就会自动创建一个组件对象(组件类的实例),并为其设置好相应的属性初值,这是在设计阶段用类创建对象的可视化操作。当然,在程序运行时也可以通过有关语句动态地创建一个对象。

组件板的每一页上都有一个指针按钮(左侧),可用于取消已选定的组件。如果选择了一个组件,但又想放弃组件的选取,只需按这个指针按钮即可。

利用菜单项 Component | Configure Palette..., 可以定制组件板的布局和内容。

Delphi XE8 的中间部分就是编辑器窗口,这个窗口不再像老版本(如 Delphi 7)那样是可以浮动的,而是将编辑器窗口作为主窗口的一部分。默认时显示 Delphi 的欢迎页面(Welcome Page),上面是任务导航链接和一些简单的帮助信息。在设计阶段,编辑器窗口就是开发人员进行用户界面设计和可视化操作的主要场地。

编辑器窗口包括窗口设计器和代码编辑器,可以查看界面设计和其对应的单元代码。通过窗口底端的标签页切换,如单击 Code 标签页,显示单元代码编辑窗口,代码编辑窗口为开发人员提供了一个输入、编辑、调试程序代码的环境,同时也是 Delphi 根据用户的可视化操作而自动生成代码的地方;单击 Design 标签页,显示设计窗口,可以在设计窗口内创建一个窗体,在窗体上可以建立各种组件对象,并快速建立一个应用界面的原型。

单元代码编辑窗口和设计窗口分别如图 1-4 和图 1-5 所示。

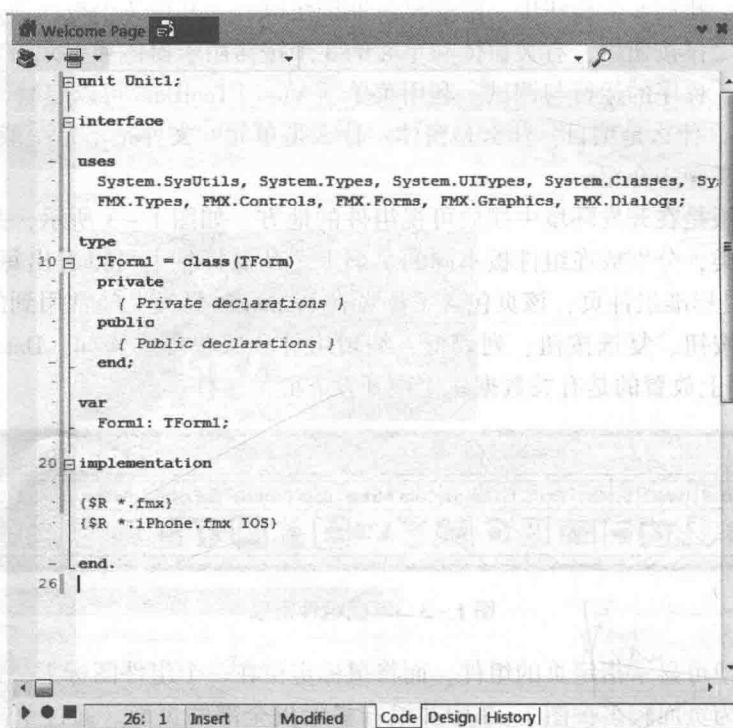


图 1-4 单元代码编辑窗口

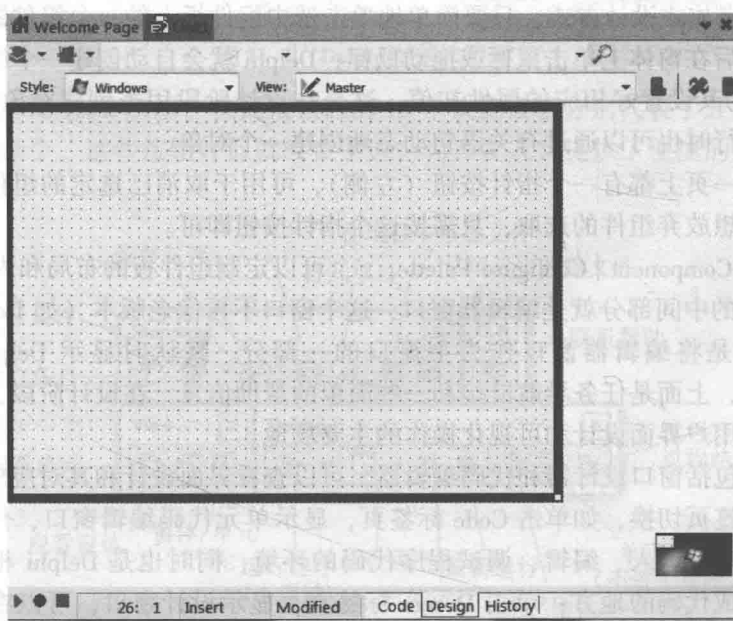


图 1-5 设计窗口

代码编辑器类似于一个多页文档（图1-4中只含一页），每一页对应一个源代码模块或文件。当在应用程序中新增一个窗体时，Delphi就会自动创建一个新的单元，并将单元名写在单元所在页的页标签中。

Delphi 代码编辑器是一种高性能的编辑器，它能用彩色亮条标出编译出错的语句，并提供了方便的代码调试功能。Delphi 代码编辑器更具特色的是被称为“代码洞察”的功能，它能做到以下几点：

- (1) 自动提示和添加各种控制语句结构。
- (2) 自动提示对象的属性和方法清单。
- (3) 自动提示函数、过程和方法的形式参数定义。

有关这方面的详细内容，稍后结合例子再做介绍。

代码浏览窗口以一种树状视图的方式显示了代码编辑器中单元文件的组成元素。如不需要，也可关闭代码浏览器，以扩大编辑窗口。

在一般情况下，窗体和单元编辑窗口占据了屏幕的大部分位置，且重叠摆放。利用快捷键（观看单元、观看窗体、单元/窗体切换）可以将编辑窗口或窗体激活，并显示在最前面。由于界面设计和代码编写是设计阶段两项主要的工作内容，且交替操作，所以在实际开发过程中，窗体和单元编辑窗口的切换是很频繁的。

2. 对象查看器

如果说组件板上的组件是类，那么在窗体上建立的组件对象（包括窗体本身）就是类的实例。例如，可以在窗体上建立一个 TButton 组件类的对象 Button1，Button1 是新建按钮对象的标识名。

对象查看器是在开发过程中控制对象的一个直观而方便的工具，它能让开发人员查看和修改一个已建对象的属性，查看和控制对象的事件及事件处理程序。

如图1-6所示，对象查看器中有两个页标签，分别是属性标签（Properties）和事件标签（Events）。对象查看器上方的小窗口用于显示当前对象及类型，当在窗体上选中某个对象（如按钮对象 Button1）后，该窗口就显示该对象的标识名和所属的类，而下面两个标签页中就显示这个对象的属性和事件清单。单击小窗口右边的下拉选单按钮，也可以从已建对象的列表中确定一个当前对象。

属性页便于开发人员在设计阶段查看和设置对象的属性。如图1-6（a）所示，左列是属性名（如 Caption），右列是属性值。每个组件都有自己特定的属性，如按钮对象就有 Name（对象标识名）、Caption（文字标题）、Left（水平位置）、Top（垂直位置）、Height（高度）、Width（宽度）、Font（字体）等属性。

虽然一个对象的属性很多，但需要开发人员必须过问的属性并不多。很多属性都可以采用系统设定的默认值，而有些属性的值（如按钮的大小、位置等属性）又可以通过可视化操作来完成（如利用鼠标拖动改变按钮大小及在窗体上的位置）。在开发过程中，若要填写或修改一个属性值，只要选择一个属性后，就可以在输入框中输入新的值。例如，选择 Caption 属性后，在右边就可以将原有的值 Button1 改为 Exit。

注意：很多属性值并不需要开发人员直接输入，而是通过下拉选单选择输入。由于这些

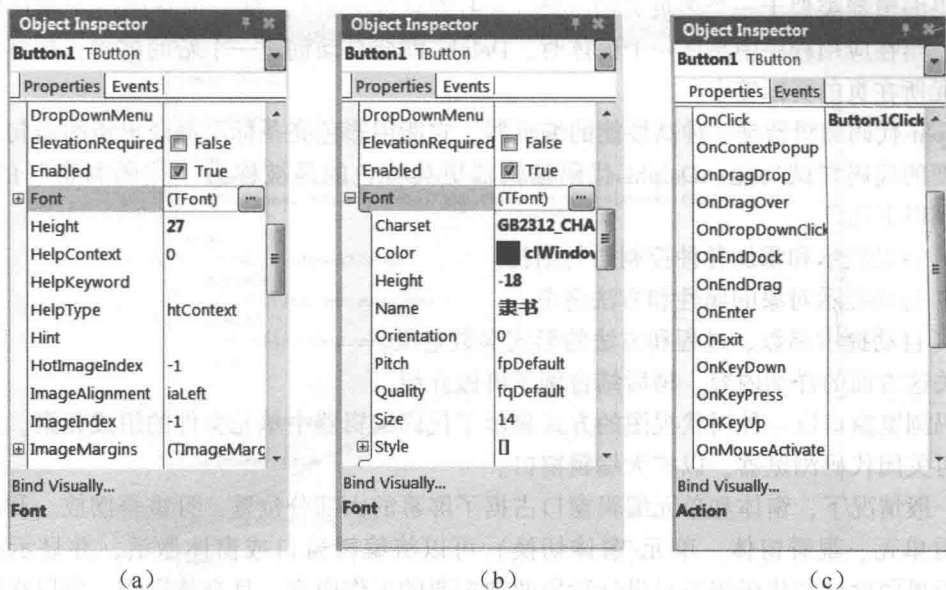


图 1-6 对象查看器

(a) 属性页例; (b) 子属性例; (c) 事件页例

属性都有自己的取值范围，如字体、颜色、对齐方式等，通过下拉选单输入不但准确无误，而且简化了输入操作。

在输入属性值时，要注意属性的数据类型。有些属性是基本数据类型，可以输入一个数字值、一个字符串或一个逻辑值（True 或 False）。但也有一些属性的数据类型是一个对象类型，即该属性又具有自己的属性（子属性）和方法。例如，图 1-6（a）中的字体属性（Font）就是 TFont 类型。

在 Delphi 中，标识名中以“T”开头的都是类的名字，开发人员也应遵循这个约定。

那么读者会问，带有子属性的属性值如何填写呢？一般带有子属性的属性名前都标有“+”，如 Font 属性前就标有“+”号，单击后，就可以展开下面的子属性，而此时“+”号也变成了“-”号。图 1-6（b）显示了 Font 属性的子属性，如有颜色（Color）、字体（Name）、大小（Size）、字型（Style）等，其中，字型子属性（Style）还有自己的子属性（粗体、斜体、下划线等）。展开子属性后，就可以分别填写这些子属性的值。另外，读者可能注意到，Font 属性值的输入框边有一个省略号按钮 ，单击后，系统会弹出一个对话框，如图 1-7 所示。在这个对话框中，开发人员也可以为各个子属性赋值。

凡是带有省略号按钮  的属性，都具有比较复杂的属性值。单击省略号按钮 ，就会弹出一个对话框或编辑框，为开发人员输入属性值提供一个专门的界面。

事件页可以让开发人员查看对象响应的事件和已编写的事件处理程序。如图 1-6（c）所示，事件页左列是事件名，右列是事件处理程序名。从图中可以看到，按钮对象所能响应的事件有 OnClick 事件（单击按钮事件）、OnEnter 事件、OnExit 事件等。图中显示只有

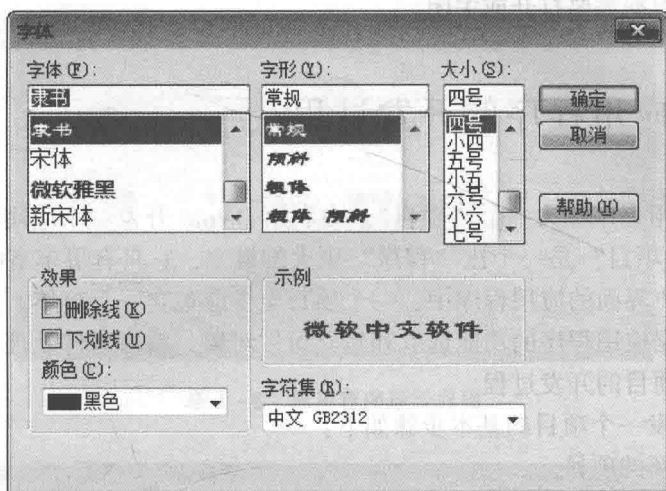


图 1-7 “字体”对话框

OnClick 事件编写了事件处理程序 Button1Click。也就是说，当应用程序运行时，如果用户单击这个按钮，Button1Click 事件处理程序就会被触发，而发生在该按钮上的其他事件一概没有反应。

3. 对象结构视图

对象结构视图之前也被称为对象树状视图 (Object Tree View)，用来以树状结构显示一个应用中已经建立的对象，以及它们之间的逻辑关系，如父子关系（包含与被包含，类似于文件夹与文件）、兄弟关系（并列，类似于同一文件夹中的两个文件）。

在 Delphi 中，有的组件扮演一种容器的角色（如窗体 Form、组件面板 Panel 等），它们专用于摆放其他组件对象，在一个容器中建立的对象与容器对象就构成了父子关系。例如，在窗体 Form1 中建立两个按钮对象 Button1 和 Button2，这两个按钮对象与窗体对象的关系都是父子关系，而这两个按钮对象的关系就是兄弟关系。

容器可用来组织对象，并能定义所包含对象通用的默认属性值。

在对象结构视图中，还可以通过可视化操作来改变对象相互之间的关系，这也类似于 Windows 系统的资源管理器中拖动文件改变位置的操作。

对象结构视图可以根据需要打开或关闭。

4. 项目管理器

在 Delphi 7 中，默认是不显示项目管理器的，而在 Delphi XE 系列中，其默认显示于窗口右上方。项目管理器的主要作用是管理每个“项目”（项目的具体内容见下一章节）的文件和配置信息。增加和删除单元文件、二进制文件及其他的项目文件，都需要通过项目管理器来进行。

5. 工具面板

Delphi XE 8 将 Delphi 7 中的控件标签变成了工具面板，并移到了右下角，用于帮助开发人员快速找到开发过程中所需要的组件。

工具面板可以根据需要打开或关闭。

1.3 Delphi 应用程序的开发过程

一个 Delphi 应用程序称为一个“项目”。在利用 Delphi 开发一个应用程序时，首先要建立一个新的项目。“项目”是一个比“程序”更大的概念，它具有更丰富的内涵。

在包含图形用户界面的应用程序中，一个项目至少应包含一个窗体。开发人员在窗体上设计用户界面，根据应用程序的功能选取和建立组件对象、编写事件处理程序与其他程序代码，最后完成一个项目的开发过程。

利用 Delphi 开发一个项目的基本步骤如下：

- (1) 建立一个新的项目。
- (2) 建立窗体。
- (3) 在窗体上建立对象。
- (4) 检查和设置对象属性。
- (5) 编写对象的事件处理程序。
- (6) 编译、试运行以及进行代码调试。
- (7) 保存项目文件，退出 Delphi 环境。

1.4 例子：第一个 Delphi 程序

本节拟通过一个实例，讲述用 Delphi 开发一个项目的基本过程和所涉及的一些基本概念，并使读者进一步熟悉 Delphi 的集成开发环境。

在本例中，界面上设置了两个输入框和一个输出框（如图 1-8 所示）。用户在两个输入框中分别输入账号和密码，然后单击“显示”按钮，显示框中显示刚输入的账号、密码信息。

单击“清除”按钮，将清空输入框、输出框。单击“退出”按钮，将结束并退出该窗体。

另外，“密码”输入框要采用屏蔽字符（#）屏蔽输入时的显示。

下面是这个例子中开发的具体步骤。

1. 建立一个新的项目

Delphi XE8 启动后，并不像 Delphi 7 那样会自动创建一个窗口程序，而是需要进行如图 1-9 所示的操作。依次单击 File | New | VCL Forms Application - Delphi 才能新建一个窗口程序。

系统会为用户建立一个新的项目 Project1，同时建立了项目的第一个窗体“Form1”。

2. 建立窗体

如上所述，系统在创建一个新项目时会自动建立项目的第一个窗体。如果一个项目包含多个窗体，则可以利用 File | New/VCL Form - Delphi 命令建立一个新的窗体。含有多个窗体