

## <<深入浅出 Delphi 6>>

### 图书基本信息

书名：<<深入浅出 Delphi 6>>

13位ISBN编号：9787900643254

10位ISBN编号：7900643257

出版时间：2002-9-1

出版时间：清华大学出版社

作者：董良,朱明峰,高磊,高忠涛

页数：794

字数：1177000

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## <<深入浅出 Delphi 6>>

### 内容概要

Delphi 6是美国Borland公司开发的可视化编程平台，它以强大的数据库功能、丰富多彩的网络程序开发能力、令人震撼的跨平台开发特性迅速成为令人瞩目的开发平台。由于引入了大量的新特性，初学者对Delphi 6的掌握会有一些困难。

本书是Delphi 6的普及与提高书籍。

本书的主要特色在于：利用丰富的实例引导读者学习Delphi的开发，并且在程序开发的过程中贯穿了面向对象的思想，用浅显的语言渗透了面向对象的理论知识。

本书对面向对象知识的全面介绍在同类书籍中是比较少见的。

另外，本书还包含了编者多年的开发经验与技巧。

全书共20章，分为6个部分。

包括：Delphi 6的新特性和Delphi的编程环境；Delphi的语法和面向对象基础；常规Windows程序开发；数据库程序开发；网络程序开发；高级程序开发技巧。

配书光盘内容为本书所涉及的实例的源代码和作者设计的练习题。

本书内容覆盖面较广，语言通俗易懂，内容深入浅出，既可作为Delphi的初学者和相关开发人员的参考书籍，又可作为广大编程爱好者和工程技术人员的参考手册。

## &lt;&lt;深入浅出 Delphi 6&gt;&gt;

## 书籍目录

## 第1部分 Delphi预备知识

## 第1章 Delphi 6概述

## 1.1 概述

## 1.1.1 Delphi的特点

## 1.1.2 Delphi 6的不同版本

## 1.1.3 Delphi 6的新特性

## 1.2 安装Delphi 6

## 1.2.1 Delphi 6的系统需求

## 1.2.2 安装Delphi 6

## 1.2.3 卸载Delphi 6

## 1.3 启动Delphi 6

## 1.3.1 启动Delphi 6

## 1.3.2 Delphi 6界面介绍

## 1.4 用Delphi 6开发控制台程序

## 1.5 Delphi 6的可视化编程

## 1.5.1 Delphi 6可视化编程的基本概念

## 1.5.2 简单的编程实例

## 1.6 本章小结

## 第2章 Delphi 6的集成开发环境

## 2.1 认识Delphi 6的集成开发环境

## 2.1.1 对象查看器

## 2.1.2 对象树状浏览器

## 2.1.3 代码编辑器

## 2.1.4 Code Insight工具

## 2.1.5 工程管理器(The Project Manager)

## 2.2 从一个小程序开始

## 2.2.1 新建一个应用程序(Application)

## 2.2.2 在窗体上添加组件(components)

## 2.2.3 为菜单和工具栏添加功能

## 2.2.4 为动作管理器添加动作

## 2.2.5 为动作管理器添加标准动作

## 2.2.6 为图像列表添加图片

## 2.2.7 添加菜单

## 2.2.8 添加工具栏

## 2.2.9 编写一个事件句柄

## 2.2.10 为New命令创建事件句柄

## 2.2.11 为Open命令创建事件句柄

## 2.2.12 为Save命令编写事件句柄

## 2.2.13 为SaveAs命令创建事件句柄

## 2.2.14 创建帮助文件

## 2.2.15 为Help Content命令创建事件句柄

## 2.2.16 为Help Index命令创建事件句柄

## 2.2.17 创建About窗口

## 2.2.18 完成程序

## 2.3 本章小结

## <<深入浅出 Delphi 6>>

### 第2部分 面向对象和Object Pascal语法

#### 第3章 从Pascal 到Object Pascal

##### 3.1 程序的组织结构

##### 3.2 注释

##### 3.3 简单数据类型

###### 3.3.1 类型的比较

###### 3.3.2 字符

###### 3.3.3 字符串

###### 3.3.4 变体类型

##### 3.4 自定义数据类型

###### 3.4.1 集合类型

###### 3.4.2 动态数组

###### 3.4.3 函数类型(function types)

##### 3.5 常量声明

###### 3.5.1 真实常量

###### 3.5.2 类型常量

##### 3.6 类型别名

##### 3.7 类型转换和强制类型转换

##### 3.8 新的过程和函数特征

###### 3.8.1 圆括号

###### 3.8.2 重载(Overloading)

###### 3.8.3 默认值参数

##### 3.9 嵌入式汇编指令

##### 3.10 位运算

##### 3.11 本章小结

#### 第4章 面向对象的基础知识

##### 4.1 从结构化到面向对象

##### 4.2 类

##### 4.3 对象

###### 4.3.1 声明和实例化

###### 4.3.2 析构

###### 4.3.3 类与对象的关系

##### 4.4 方法

###### 4.4.1 方法和普通函数的比较

###### 4.4.2 Self变量

###### 4.4.3 类方法

###### 4.4.4 函数型类型

##### 4.5 封装性

###### 4.5.1 类的成员

###### 4.5.2 类的属性

###### 4.5.3 可见性指示符

##### 4.6 继承性

###### 4.6.1 继承的概念

###### 4.6.2 基类

###### 4.6.3 继承的可见性

###### 4.6.4 属性的继承

###### 4.6.5 方法的继承

## &lt;&lt;深入浅出 Delphi 6&gt;&gt;

- 4.7 多态性
- 4.8 as、is运算符
- 4.9 异常
  - 4.9.1 Delphi是如何处理错误的
  - 4.9.2 处理异常
  - 4.9.3 异常的嵌套
  - 4.9.4 触发异常
  - 4.9.5 异常的调试
  - 4.9.6 使用异常对象
- 4.10 类引用(class-reference)
- 4.11 Delphi 6的类库介绍
  - 4.11.1 VCL---可视化组件库
  - 4.11.2 CLX---跨平台组件库
- 4.12 Delphi标准开发文档
  - 4.12.1 文件命名规则
  - 4.12.2 源代码书写规则
  - 4.12.3 类及变量命名规则
- 4.13 本章小结
- 第3部分 Windows 程序开发初步
- 第5章 过渡到Delphi 6
  - 5.1 DOS程序过渡到Windows程序
    - 5.1.1 DOS平台和Windows平台的差异
    - 5.1.2 DOS程序和Windows程序的差异
    - 5.1.3 从结构化走向面向对象
  - 5.2 C / C++过渡到面向对象的Pascal
    - 5.2.1 语法区别
    - 5.2.2 函数的区别
    - 5.2.3 类型的转换
  - 5.3 从VB到Delphi
    - 5.3.1 Delphi的武器--双平台开发
    - 5.3.2 Delphi与VB相比的优点
    - 5.3.3 Delphi与VB的语法对比
  - 5.4 本章小结
- 第6章 组件的使用
  - 6.1 什么是组件
  - 6.2 理解组件库
    - 6.2.1 VCL与CLX
    - 6.2.2 组件的实质--类
    - 6.2.3 组件的Object Pascal语言描述
    - 6.2.4 属性、方法和事件
  - 6.3 控件的常用属性和事件
    - 6.3.1 控件的基本属性
    - 6.3.2 控件的标准事件
  - 6.4 组件的使用
    - 6.4.1 解释Delphi的组件面板
    - 6.4.2 组件的一般使用方法
    - 6.4.3 对组件进行操作

## &lt;&lt;深入浅出 Delphi 6&gt;&gt;

- 6.4.4 修改组件属性的一般方法
- 6.4.5 方法的调用
- 6.4.6 事件的处理
- 6.4.7 解析一个Delphi程序
- 6.5 Windows应用程序的基础--窗体
  - 6.5.1 程序的主窗体
  - 6.5.2 向工程中添加新的窗体
  - 6.5.3 隐藏主窗体
  - 6.5.4 模态窗体与非模态窗体
  - 6.5.5 窗体的常用属性、方法和事件
  - 6.5.6 一个窗体使用的例子
- 6.6 文本输入组件
  - 6.6.1 TEdit与TMaskEdit控件
  - 6.6.2 TMemo与TRichEdit控件的常用属性
  - 6.6.3 RichEdit(控件只适用于VCL)
  - 6.6.4 TLabeledEdit控件
- 6.7 特殊输入组件
  - 6.7.1 滚动条组件(Scroll bars)
  - 6.7.2 Trackbar组件
  - 6.7.3 Up-Down控件(仅适用于VCL)
  - 6.7.4 SpinEdit控件(仅适用于CLX)
  - 6.7.5 Hotkey组件(仅适用于VCL)
  - 6.7.6 Splitter组件
  - 6.7.7 特殊输入组件应用实例
- 6.8 按钮类组件
  - 6.8.1 TButton组件
  - 6.8.2 位图按钮(TBitBtn)
  - 6.8.3 加速按钮组件(Speed buttons)
  - 6.8.4 复选框按钮(TCheckBox)
  - 6.8.5 单选按钮(TRadioButton)
- 6.9 列表组件
  - 6.9.1 列表框及复选按钮列表框组件(TListBox与TCheckListBox)
  - 6.9.2 组合框组件(TComboBox)
  - 6.9.3 ImageList组件
  - 6.9.4 扩展组合框组件(TComboBoxEx)
  - 6.9.5 颜色选择组合框(TColorBox)
  - 6.9.6 TreeView组件(TTreeView)
  - 6.9.7 ListView组件
  - 6.9.8 日期 / 时间选取组件以及日历组件(仅适用于VCL)
- 6.10 分组组件
  - 6.10.1 普通分组组件
  - 6.10.2 TScrollBox组件
  - 6.10.3 标签组件(TTabControl)
  - 6.10.4 分页组件(TPageControl)
  - 6.10.5 标题栏组件(THeaderControl)
  - 6.10.6 应用举例：TPageControl组件的使用
- 6.11 信息反馈组件

## &lt;&lt;深入浅出 Delphi 6&gt;&gt;

6.11.1 标签及静态文本组件(TLabel and TStaticText)

6.11.2 状态栏组件

6.11.3 进度条组件

6.11.4 信息反馈组件应用实例

6.12 表格显示组件

6.12.1 DrawGrid组件

6.12.2 字符表格组件StringGrid

6.12.3 ValueListEditor组件(仅用于VCL)

6.13 图形显示组件

6.13.1 Image组件

6.13.2 几何图形组件(TShapes)

6.13.3 Bevel组件

6.13.4 PaintBox组件(TPaintBox)

6.13.5 动画组件(TAnimate)

6.14 菜单的创建与管理

6.14.1 向窗体中添加菜单组件

6.14.2 菜单编辑器(Menu Designer)

6.14.3 定制菜单

6.14.4 使用菜单编辑器的上下文关联菜单

6.14.5 在窗体中使用菜单

6.14.6 在窗体中动态创建菜单

6.14.7 菜单的应用实例

6.15 工具栏的使用

6.15.1 使用工具栏组件ToolBar

6.15.2 CoolBar组件

6.16 对话框组件

6.16.1 打开标准对话框

6.16.2 对话框的属性和使用

6.16.3 调用MessageBox函数实现对话框

6.17 组件的其他应用

6.17.1 操作控件中的文本

6.17.2 使用ActionList管理菜单和工具栏

6.18 本章小结

第7章 常见编程方法与技巧

7.1 字符串的操作

7.2 文件的操作

7.2.1 文件的类型及声明

7.2.2 类型文件的操作

7.2.3 文本文件的操作

7.2.4 无类型文件的操作

7.2.5 使用文件流

7.2.6 有关文件名和路径的操作

7.3 目录和驱动器的操作

7.4 多媒体技术

7.4.1 操作位图

7.4.2 操作JPEG文件

7.4.3 操作画布

## &lt;&lt;深入浅出 Delphi 6&gt;&gt;

- 7.4.4 特殊显示效果
- 7.4.5 音频、视频
- 7.5 打印
  - 7.5.1 简单打印
  - 7.5.2 常规打印
  - 7.5.3 打印相关API函数
- 7.6 注册表操作
- 7.7 板与数据交换
- 7.8 拖放(Drag & Drop)操作
  - 7.8.1 启动一个拖动操作
  - 7.8.2 接受被拖动的项目
  - 7.8.3 放置项目
  - 7.8.4 结束一个拖动操作
  - 7.8.5 用拖动对象自定义施放操作
  - 7.8.6 改变拖动鼠标指针
- 7.9 拖放与融合(Drag & Dock)操作
  - 7.9.1 使窗口化控件作为泊点
  - 7.9.2 使控件作为可泊入子控件
  - 7.9.3 控制子控件如何泊入
  - 7.9.4 控制子控件如何移出
  - 7.9.5 控制子控件如何响应drag-and-dock操作
- 7.10 程序启动参数
- 7.11 组件的安装与卸载
- 7.12 本章小结
- 第8章 多线程应用程序
  - 8.1 进程与线程
    - 8.1.1 进程
    - 8.1.2 线程
  - 8.2 多线程程序的预备知识
  - 8.3 Tthread类
    - 8.3.1 创建线程对象
    - 8.3.2 TThread的属性和方法
  - 8.4 线程的优先级
  - 8.5 线程中的变量存储
  - 8.6 Synchronize同步函数
  - 8.7 多线程程序的调试
  - 8.8 一个演示程序
  - 8.9 本章小结
- 第9章 动态链接库
  - 9.1 动态链接库简介
  - 9.2 编写动态链接库
    - 9.2.1 创建一个动态链接库
    - 9.2.2 动态链接库的结构
    - 9.2.3 Exports语句
    - 9.2.4 动态链接库初始化和退出
  - 9.3 调用DLL
    - 9.3.1 静态加载

## &lt;&lt;深入浅出 Delphi 6&gt;&gt;

- 9.3.2 动态加载
- 9.4 Windows下的共享内存管理
- 9.5 DLL中的异常
- 9.6 调用约定
- 9.7 本章小结
- 第10章 调试程序
- 10.1 常见的编程错误
  - 10.1.1 语法错误
  - 10.1.2 指针错误
  - 10.1.3 面向对象编程中的错误
- 10.2 使用集成调试器
- 10.3 本章小结
- 第4部分 数据库开发
- 第11章 数据库程序入门
- 11.1 关系型数据库简介
- 11.2 Delphi与数据库特性
  - 11.2.1 数据库相关组件简介
  - 11.2.2 数据库类型
  - 11.2.3 数据库安全
  - 11.2.4 事务
  - 11.2.5 参照完整性、存储过程和触发器
- 11.3 Delphi的数据库架构
  - 11.3.1 数据库架构
  - 11.3.2 直接连接数据库服务器
  - 11.3.3 使用磁盘上的专用文件
  - 11.3.4 连接其他数据集
- 11.4 数据控件
  - 11.4.1 使用数据控件
  - 11.4.2 普通数据控件特性
  - 11.4.3 连接数据控件和数据集
  - 11.4.4 运行期改变相连数据集
  - 11.4.5 连接数据源
  - 11.4.6 响应数据源发生的变化
  - 11.4.7 编辑和更新数据
  - 11.4.8 使用户能够在控件中编辑数据
  - 11.4.9 使用控件编辑数据
  - 11.4.10 控制数据显示
  - 11.4.11 刷新数据
  - 11.4.12 允许键盘、鼠标和Timer事件
  - 11.4.13 选择如何组织数据
  - 11.4.14 显示一个记录
  - 11.4.15 将数据以标签方式显示
  - 11.4.16 在编辑框中显示和编辑字段
  - 11.4.17 在memo控件中显示和编辑文本
  - 11.4.18 在富文本编辑框显示和编辑文本数据
  - 11.4.19 在image控件中显示和编辑图形字段
  - 11.4.20 在列表框和组合框中显示和编辑数据

## &lt;&lt;深入浅出 Delphi 6&gt;&gt;

- 11.5 TDataSource组件介绍
  - 11.5.1 TDataSource的继承关系
  - 11.5.2 TDataSource组件的重要属性
  - 11.5.3 TDataSource组件的重要方法
  - 11.5.4 TDataSource组件的重要事件
  - 11.5.5 应用程序中建立TDataSource组件的基本步骤
- 11.6 一个简单的数据库程序
  - 11.6.1 建立一个新应用程序
  - 11.6.2 设置属性值
  - 11.6.3 在Form上添加对象
  - 11.6.4 连接数据库
- 11.7 本章小结
- 第12章 BDE的使用
  - 12.1 BDE
  - 12.2 BDE结构框架
    - 12.2.1 基于BDE的架构
    - 12.2.2 使用BDE支持的数据集组件
    - 12.2.3 用数据库和会晤与数据集建立关联
    - 12.2.4 缓存存储BLOB类型数据
    - 12.2.5 使用BDE句柄属性
  - 12.3 BDE组件及其使用
    - 12.3.1 BDE组件简介
    - 12.3.2 使用TTable
    - 12.3.3 使用TQuery
    - 12.3.4 使用TStoredProc控件
    - 12.3.5 使用IDatabase组件
    - 12.3.6 TSession组件
    - 12.3.7 TBatchMove组件
    - 12.3.8 TUpdateSQL组件介绍
    - 12.3.9 TNestedTable组件
    - 12.3.10 TBDEClientDataSet组件介绍
  - 12.4 本章小结
- 第13章 ADO的使用
  - 13.1 认识ADO
  - 13.2 ADOExpress
    - 13.2.1 TADODConnection
    - 13.2.2 TADODCommand组件
    - 13.2.3 TADODDataSet组件
    - 13.2.4 TADODTable
    - 13.2.5 通过ADO连接
    - 13.2.6 应用ADO开发Master/Detail应用程序
  - 13.3 本章小结
- 第14章 开发跨平台数据库
  - 14.1 单向数据集简介
  - 14.2 单向数据集的种类
  - 14.3 连接数据库服务器
  - 14.4 指定显示何种数据

## &lt;&lt;深入浅出 Delphi 6&gt;&gt;

- 14.4.1 显示查询结果
- 14.4.2 显示表中记录
- 14.4.3 显示存储过程的结果
- 14.5 取得数据
  - 14.5.1 准备数据集
  - 14.5.2 获取多个数据集
- 14.6 执行不返回记录的命令
  - 14.6.1 指定要执行的命令
  - 14.6.2 执行命令
  - 14.6.3 创建和修改服务器的元数据
- 14.7 建立主从关联游标
- 14.8 调用模式信息
  - 14.8.1 将元数据装入单向数据集
  - 14.8.2 元数据数据集的结构
- 14.9 本章小结
- 第5部分 网络和服务器开发
- 第15章 开发丰富多彩的网络程序
  - 15.1 网络程序开发简介
  - 15.2 Internet页控件
  - 15.3 FastNet页控件简介
  - 15.4 Indy控件群简介
    - 15.4.1 Indy Clients
    - 15.4.2 Indy Servers
    - 15.4.3 Indy Misc
  - 15.5 做一个简单的浏览器
  - 15.6 写自己的E-mail客户端程序
  - 15.7 创建一个FTP客户端程序
  - 15.8 Ping和Tracert命令
  - 15.9 本章小结
- 第16章 开发基于套接字的网络程序
  - 16.1 Socket简介
  - 16.2 套接字连接种类
  - 16.3 有关套接字的组件
  - 16.4 服务器和客户端
  - 16.5 一个TCP协议的聊天程序
  - 16.6 本章小结
- 第17章 开发Web服务器应用程序
  - 17.1 网络编程的挑战
  - 17.2 使用Web Broker技术
    - 17.2.1 用Web Broker创建Web服务器应用程序的方法
    - 17.2.2 Web服务器应用程序的结构
    - 17.2.3 Web分派器
    - 17.2.4 请求消息分派
    - 17.2.5 动作项
    - 17.2.6 生成响应信息
    - 17.2.7 利用组件生成响应信息内容
  - 17.3 使用WebSnap技术

## &lt;&lt;深入浅出 Delphi 6&gt;&gt;

- 17.3.1 WebSnap的新特性
- 17.3.2 WebSnap基础组件
- 17.3.3 创建一个新的应用程序
- 17.3.4 创建一个CountryTable页面
- 17.3.5 在CountryTable模块中增加数据组件
- 17.3.6 创建表格显示数据
- 17.3.7 增加一个编辑窗体
- 17.3.8 运行完整的程序
- 17.3.9 错误报告
- 17.4 本章小结
- 第18章 高级网络编程技术
- 18.1 使用XML文档
- 18.1.1 XML文档简介
- 18.1.2 使用文档对象模型
- 18.1.3 使用XML组件
- 18.1.4 使用TXMLDocument控件
- 18.1.5 使用XML节点
- 18.1.6 使用数据绑定向导 ( Data Binding wizard ) 提炼XML文档
- 18.1.7 数据绑定向导 ( Data Binding wizard ) 的使用
- 18.1.8 使用XML数据绑定向导产生的代码
- 18.2 使用Web服务
- 18.2.1 Web服务简介
- 18.2.2 编写支持Web服务的服务器程序
- 18.2.3 定义可调用接口
- 18.2.4 在可调用接口中使用复杂类型
- 18.2.5 创建并注册实现
- 18.2.6 为Web服务建立自定义异常处理类
- 18.2.7 为一个Web服务程序产生WSDL文档
- 18.2.8 编写SOAP客户端
- 18.3 本章小结
- 第6部分 高级开发
- 第19章 深入Windows编程
- 19.1 调用Windows API
- 19.1.1 API函数的分类
- 19.1.2 参数类型的转换
- 19.1.3 调用API
- 19.2 Windows中的消息
- 19.2.1 理解消息
- 19.2.2 消息的类型
- 19.2.3 Windows消息系统是如何工作的
- 19.2.4 Delphi的消息系统
- 19.2.5 消息处理
- 19.2.6 发送自己的消息
- 19.2.7 VCL内部的消息
- 19.2.8 用户自定义的消息
- 19.2.9 消息与事件之间的关系
- 19.3 使用挂钩 ( Hook )

## &lt;&lt;深入浅出 Delphi 6&gt;&gt;

- 19.3.1 什么是挂钩
- 19.3.2 设置挂钩
- 19.3.3 使用挂钩函数
- 19.3.4 使用脱钩函数
- 19.3.5 记录键盘鼠标动作与回放：一个挂钩实例
- 19.4 程序中使用汇编语言
  - 19.4.1 嵌入汇编的两种形式
  - 19.4.2 Delphi中的汇编
  - 19.4.3 简易的参数访问
  - 19.4.4 var声明的参数
  - 19.4.5 Register调用约定
  - 19.4.6 全汇编过程
  - 19.4.7 汇编语言访问记录
  - 19.4.8 简单的例子
- 19.5 使用托盘图标
  - 19.5.1 API
  - 19.5.2 第一步：使用TrayIcon
  - 19.5.3 第二步：在TrayIcon上打开弹出式菜单
  - 19.5.4 再进一层：维护多个TrayIcon
- 19.6 COM、COM+与MTS的简单知识
- 19.7 本章小结
- 第20章 组件开发技术
  - 20.1 组件创建简介
    - 20.1.1 组件创建概述
    - 20.1.2 可视化组件库（VCL）
    - 20.1.3 组件和类
    - 20.1.4 怎样创建组件
    - 20.1.5 组件有什么要求
    - 20.1.6 创建新组件
    - 20.1.7 测试未安装的组件
    - 20.1.8 测试已安装的组件
  - 20.2 组件开发人员的面向对象编程
    - 20.2.1 组件开发人员的面向对象编程概述
    - 20.2.2 定义新类
    - 20.2.3 派生新类
    - 20.2.4 声明新的组件类
    - 20.2.5 祖先和子孙
    - 20.2.6 控制访问
    - 20.2.7 分派方法
    - 20.2.8 类和指针
  - 20.3 创建属性
    - 20.3.1 创建属性概述
    - 20.3.2 为什么创建属性
    - 20.3.3 属性类型
    - 20.3.4 公开继承属性
    - 20.3.5 定义组件属性
    - 20.3.6 创建数组属性

## &lt;&lt;深入浅出 Delphi 6&gt;&gt;

20.3.7 为子组件创建属性

20.3.8 为接口创建属性

20.3.9 存储和加载属性

20.4 创建事件

20.4.1 创建事件概述

20.4.2 什么是事件

20.4.3 实现标准事件

20.4.4 定义自己的事件

20.5 创建方法

20.5.1 创建方法概述

20.5.2 消除依赖性

20.5.3 命名方法

20.5.4 保护方法

20.5.5 使方法成为虚方法

20.5.6 声明方法

20.6 处理消息

20.6.1 处理消息概述

20.6.2 理解消息处理系统

20.6.3 改变消息处理

20.6.4 创建新的消息句柄

20.7 使组件在设计阶段可用

20.7.1 概述

20.7.2 注册组件

20.7.3 添加面板位图

20.7.4 为组件提供帮助

20.7.5 添加属性编辑器

20.7.6 属性分类

20.7.7 添加组件编辑器

20.7.8 编译组件并打包

20.8 修改现有组件

20.8.1 修改现有组件概述

20.8.2 创建并且注册组件

20.8.3 修改组件类

20.9 本章小结

附录A Delphi附带工具的介绍

附录B Delphi函数、属性速查表

附录C 错误信息与异常

附录D 配书光盘说明

<<深入浅出 Delphi 6>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介, 请支持正版图书。

更多资源请访问:<http://www.tushu007.com>