

万水

编程实例五十讲丛书

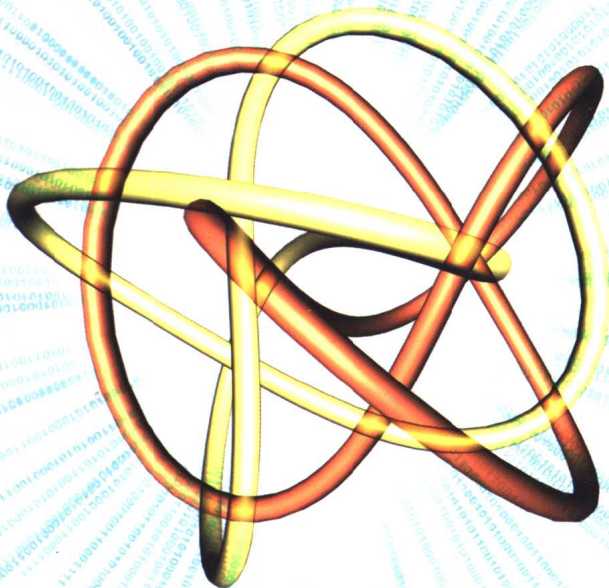
# 突破

# Delphi 7.0

## 编程实例

## 五十讲

张增强 等编著



中国水利水电出版社  
www.waterpub.com.cn

万水编程实例五十讲丛书

# 突破 Delphi 7.0 编程实例五十讲

张增强 等编著

中国水利水电出版社

## 内 容 提 要

Delphi 是面向对象的可视化编程语言,它是目前面向对象和控件开发性能最好的工具之一。它所提供的大量可重用控件和用户自建模板技术,极大地提高了应用系统的开发速度。

本书融合了在 Windows 下利用 Delphi 7.0 开发应用程序的理论和实践,全面而深入地介绍了利用 Delphi 7.0 开发应用程序的常用方法和技巧。本书以实例为主线,以编程方法的介绍为基本,深入浅出地讲解了使用 Delphi 7.0 进行程序开发的各种技巧。

本书通过 50 个实例,讲解 Delphi 7.0 的开发和应用方法。全书共分 9 篇,包括用户界面设计、图形图像编程、文件管理、多媒体技术、系统控制、网络和数据库编程以及游戏制作等内容。

### 图书在版编目 (CIP) 数据

突破 Delphi 7.0 编程实例五十讲/张增强等编著. —北京:中国水利水电出版社, 2002

(万水编程实例五十讲丛书)

ISBN 7-5084-1276-1

I.突… II.张… III. 软件工具—程序设计 IV. TP311.56

中国版本图书馆 CIP 数据核字 (2002) 第 089925 号

书 名	突破 Delphi 7.0 编程实例五十讲
作 者	张增强 等编著
出版、发行	中国水利水电出版社 (北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@public3.bta.net.cn (万水) sale@waterpub.com.cn
经 售	电话: (010) 68359286 (万水)、63202266 (总机)、68331835 (发行部) 全国各地新华书店
排 版	北京万水电子信息有限公司
印 刷	北京市天竺颖华印刷厂
规 格	787×1000 毫米 16 开本 23 印张 507 千字
版 次	2003 年 1 月第一版 2003 年 1 月北京第一次印刷
印 数	0001—5000 册
定 价	32.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社发行部负责调换

版权所有·侵权必究

# 前 言

目前有众多 Windows 下的开发工具，而 Delphi 的开发环境是目前面向对象和控件开发性能最好的工具之一。它使用 Microsoft Windows 图形用户界面的许多先进特性和设计思想，采用了弹性、可重复利用、完整的、面向对象的程序语言，拥有当今世界上最快的编辑器，以及最为领先的数据库技术，功能强大。

Delphi 7.0 是 Inprise 公司的新一代面向对象、可视化应用程序的开发环境，工作在 Windows 95/98/2000/NT 操作系统上。Delphi 7.0 的基础语言是 Pascal，它继承了 Pascal 语言严谨的优点，如代码结构清晰、可读性好和代码执行效率高等。使用 Delphi 7.0 可以编写 Win32 控制台应用程序和具有图形用户界面（GUI）的 Win32 应用程序。Delphi 7.0 集成开发环境不仅提供了一整套用于设计、编写、测试、调试和发布应用程序的工具软件，同时还包含了大量可以重用的控件、模板和向导。和传统的 Windows 应用程序开发方法相比，Delphi 具有快速和真正可视化的特点。更加难能可贵的是，在 Delphi 7.0 中可以非常方便地进行多媒体、动画、网络和数据库等应用程序的开发。

本书以实例的形式向读者解析如何使用 Delphi 7.0。全书制作了 50 个实例，满足不同层次读者开发各方面应用程序的需要。每个实例都具有较强的针对性，突出讲解了 Delphi 7.0 的开发思想、技术和方法。在讲解中，把每个实例又分成若干部分，分别详细介绍。在每个实例后面都附有程序的源代码，对重要的语句都有详尽的注释，便于读者轻松掌握各个实例所涉及的相关内容。通过本书的学习，读者将以一种更为简洁、迅速的方式掌握 Delphi 7.0 编程技术。

本书内容广泛，涉及到 Windows 编程的各个方面。全书共分 9 篇。

第一篇为用户界面设计，讲述了如何使用 Delphi 开发程序，以及如何创建多文档窗体，设置特殊效果窗体等内容。

第二篇为图形图像处理，讲述了各种图形图像编程的处理技巧。

第三篇为文件管理，讲述了如何对文件进行控制，以及搜索文件，读写 INI 文件等内容。

第四篇为系统控制，讲述了如何使用 Delphi 和 Windows 系统交互，控制鼠标，设计程序托盘等内容。

第五篇为多媒体程序设计，介绍如何利用 Delphi 开发各种多媒体的应用程序。

第六篇为高级编程技巧，讲述了 OLE 对象、创建和使用对象，制作多线程应用程序等内容。

第七篇为数据库编程，讲述了开发数据库应用程序的基本方法，同时讲述了 ADO 技术、缓冲池技术以及事务处理等高级内容。

第八篇为网络编程，讲述了使用 Delphi 设计网络程序的方法。包括设计聊天室、接收电

子邮件以及制作网页浏览器等内容。

第九篇为游戏制作，讲述了两个游戏的制作过程。

本书由张增强执笔编写，此外，王瑾、吴浩、李炎、刘伟、刘华刚、朱峰、赵晓燕、李晓、马苍、郝春容、韦勇、成美华、萧峰、李菊、张浩然、李欣、张浩、李想、朱大成、杨竞锐、王贵新、张诚华和李鹤语等同志在整理材料方面给予了作者很大的帮助，在此一并致以感谢！

由于时间仓促，加之编者的水平有限，缺点和错误在所难免，恳请专家和广大读者不吝赐教，批评指正。

编者

2002年10月

# 目 录

## 前言

## 第一篇 用户界面设计

第一讲	第一个应用程序.....	2
第二讲	多文档程序.....	8
第三讲	动态菜单.....	18
第四讲	拖放列选项.....	27
第五讲	闪现窗体.....	34
第六讲	不规则窗体.....	38
第七讲	文本编辑器.....	43

## 第二篇 图形图像编程

第八讲	小画笔.....	52
第九讲	抓取屏幕.....	63
第十讲	图片的显示特效.....	71
第十一讲	屏幕保护.....	76
第十二讲	制作图表.....	83
第十三讲	OpenGL 编程.....	88
第十四讲	桌面晃动.....	96

## 第三篇 文件管理

第十五讲	文件管理器.....	105
第十六讲	利用递归法搜索目录中的文件.....	113
第十七讲	搜索文件.....	117
第十八讲	利用内存映射实现文件操作.....	123
第十九讲	读写 INI 文件.....	130

## 第四篇 系统控制

第二十讲	生成文件的快捷方式.....	136
第二十一讲	放大镜.....	140
第二十二讲	托盘程序.....	147
第二十三讲	禁止程序二次运行.....	152

第二十四讲	超级鼠标.....	155
-------	-----------	-----

## 第五篇 多媒体程序设计

第二十五讲	CD 播放器.....	165
第二十六讲	播放动画.....	175
第二十七讲	RealPlayer 音频播放器.....	181
第二十八讲	图片浏览器.....	185

## 第六篇 高级编程技巧

第二十九讲	使用 OLE 对象.....	195
第三十讲	创建和使用对象.....	203
第三十一讲	多线程应用.....	210
第三十二讲	Hook 技术.....	217
第三十三讲	自定义控件.....	224

## 第七篇 数据库编程

第三十四讲	使用 Rave 控件.....	233
第三十五讲	创建数据库.....	239
第三十六讲	图书管理系统.....	244
第三十七讲	ADO 技术.....	254
第三十八讲	捕捉异常.....	264
第三十九讲	使用 SQL 查询.....	271
第四十讲	MTS 缓冲池技术.....	278
第四十一讲	事务处理.....	285

## 第八篇 网络编程

第四十二讲	网页浏览器.....	292
第四十三讲	聊天室工具.....	300
第四十四讲	将数据库内容以 HTML 文件格式输出.....	306
第四十五讲	实现 Ping 操作.....	316
第四十六讲	获取网上邻居的工作组.....	320
第四十七讲	信使服务程序.....	324
第四十八讲	计数器.....	328

## 第九篇 游戏制作

第四十九讲	拼图游戏.....	335
第五十讲	拯救地球.....	351

# 第一篇

## 用户界面设计

### 本篇导读

一个成熟的应用程序，需要包括生动活泼的用户界面，使用户在使用程序时有一种舒适感，对程序操作没有畏惧心理。这就需要程序员在用户界面上下功夫，熟练掌握各种界面的制作方法。

本篇主要针对用户界面设计的各个方面进行讲解，主要包括如下实例：

- 第一讲 第一个应用程序
- 第二讲 多文档程序
- 第三讲 动态菜单
- 第四讲 拖放列选项
- 第五讲 闪现窗体
- 第六讲 不规则窗体
- 第七讲 文本编辑器



# 第一讲 第一个应用程序

## 【实现目标】

为了适合不同的读者，在本实例中编写了一个简单的应用程序，如图 1-1 所示。程序运行后，单击“点击我”按钮，将会在窗口中出现一行文字，显示“这是我的第一个 Delphi 程序”字样；单击“退出”按钮将会退出程序。

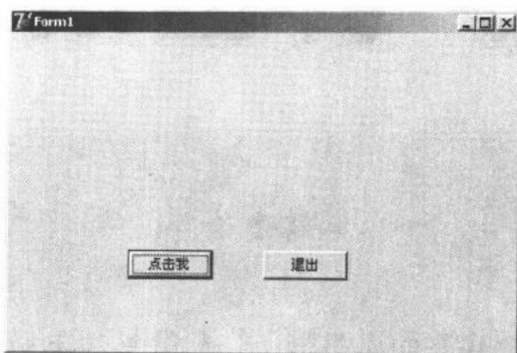


图 1-1 程序的运行结果

## 【背景知识介绍】

Delphi 实际上是 Object Pascal 语言的一种版本，它与传统的 Pascal 语言有天壤之别。一个 Delphi 程序首先是应用程序框架，而这一框架正是应用程序的“骨架”。在骨架上即使没有附着任何东西，仍可以严格地按照设计运行。默认的应用程序是一个空白的窗体 (Form)，它可以运行，结果得到一个空白的窗口，这个窗口具有 Windows 窗口的全部性质，即可以被放大缩小、移动、最大最小化等，但您却没有编写一程序。因此，可以说应用程序框架通过提供所有应用程序共有的东西，为用户应用程序的开发打下了良好的基础。Delphi 已经为您做好了一切基础工作——程序框架就是一个已经完成的可运行应用程序，只是不处理任何事情。您所需要做的，只是在程序中加入完成所需功能的代码而已。

在了解了 Delphi 7.0 的基本原理之后，用户就可以进行应用程序的开发了。Delphi 应用程序开发的基本步骤为：

- (1) 建立窗体；
- (2) 在窗体上添加控件；
- (3) 检查和设定对象属性；
- (4) 编写响应事件处理程序；
- (5) 保存文件；

(6) 编译、运行。

## 【开始工作】

### 1. 启动程序

(1) 单击 Windows 系统的开始菜单，选择“程序”→Borland Delphi 7.0→Delphi 7，如图 1-2 所示。就可以启动 Delphi 应用程序，启动后的 Delphi 窗口如图 1-3 所示。

(2) 此时程序自动生成了一个窗体，名称为 Form1，如图 1-3 所示。如果用户不小心关掉了窗体窗口，可以用 File 菜单下的 New Application 命令创建一个新的应用文件。



图 1-2 选择 Delphi 程序

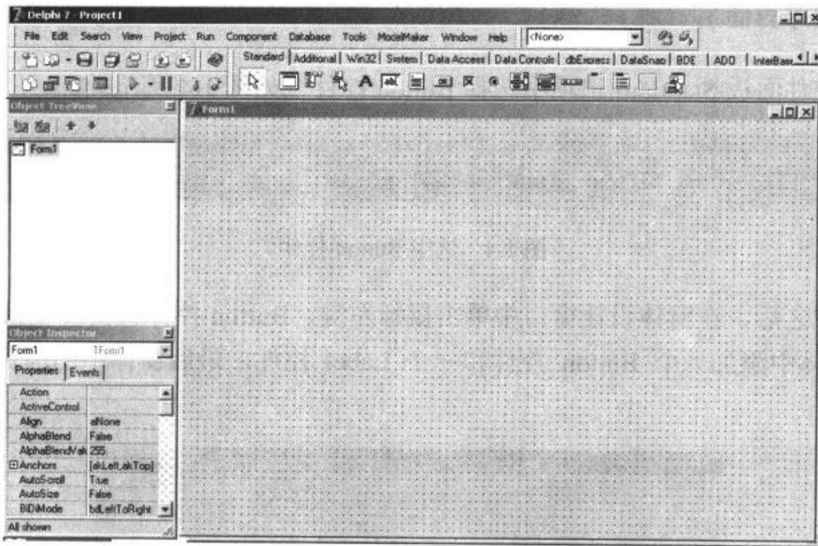


图 1-3 Delphi 7.0 的运行窗口

此时 Delphi 自动生成的代码如下：

```
unit Unit1;      //单元文件名
interface      //接口关键字，用它来标识文件所调用的单元文件
uses           //使用的公共单元
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs;
Type//定义程序所使用的控件以及控件所对应的事件
  TForm1 = class(TForm)
  Private//定义私有变量和私有过程
    { Private declarations }
  public//定义公共变量和公共过程
    { Public declarations }
```

```
end;  
var//定义程序使用的公共变量  
  Form1: TForm1;  
implementation//程序代码实现部分  
{ $R *.dfm }  
end.
```

这些自动生成的代码主要是定义一些程序的基本信息。

其中第一行指明单元的名称为 `Unit1.pas`，也就是单元将以 `Unit1.pas` 名称保存到存储器上。

`Interface` 主要是程序的接口部分，`uses` 定义了程序要使用的系统单元，比如 `Windows`、`Messages` 等单元，这些单元将在后面有所介绍。`Type` 部分主要用来声明程序所使用的公共变量以及系统控件等。这些将在第二篇详细介绍。

最后 `Implementation` 部分是程序的具体实现内容，是需要程序员来编写代码的部分。

## 2. 创建程序界面

(1) 在控件面板 `Standard` 选项中选择 `Button` 控件，如图 1-4 所示。



图 1-4 选中 Button 控件

(2) 选中之后，在窗体设计窗口中单击鼠标左键，`Button` 控件就出现在窗体上。用同样的方法再向窗体中添加一个 `Button` 控件和一个 `Label` 控件，最后窗体中的控件布置如图 1-5 所示。

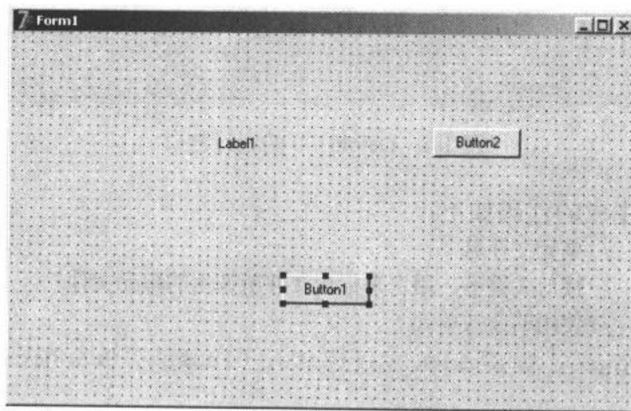


图 1-5 窗体中的控件



**提示：**Delphi 提供了“提示”功能，在选择控件时，如果用户的鼠标在某个控件上方停留一段时间，在鼠标旁会出现一个小的提示框，显示这个控件的作用。

### 3. 设定控件属性

现在摆在用户眼前的已经是一个应用程序界面的雏形了。但是用户也许并不满意这个界面，所有的按钮上面都是默认的 Button1、Button2，这对于一个应用程序的界面来说并不合适，而且也没有提供键盘快捷方式。如何改变这种情况？我们可以在对象监视器中对这个界面中控件的属性进行更改。

(1) 移动控件到合适的位置。具体的方法是：用鼠标单击要移动的控件，使其变为选中状态（控件四周有 8 个小黑矩形），然后用鼠标拖动控件就可以移动了。最后各控件的位置如图 1-6 所示。

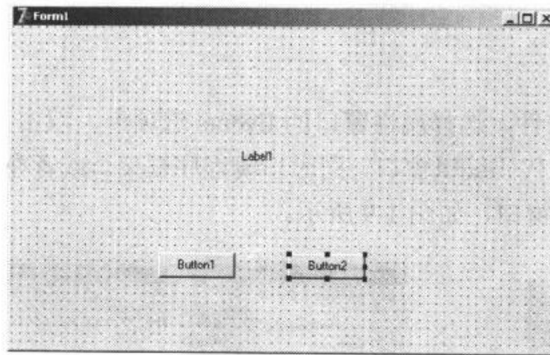


图 1-6 移动后的控件布置

(2) 选中 Button1，在对象监视器的 Properties 选项卡中将会出现和 Button1 控件相关的属性。单击 Caption 属性后面的编辑框，如图 1-7 所示。此时编辑框中的文字变为全选状态，向其中输入文字可以改变按钮的显示名称，这里改为“点击我”。

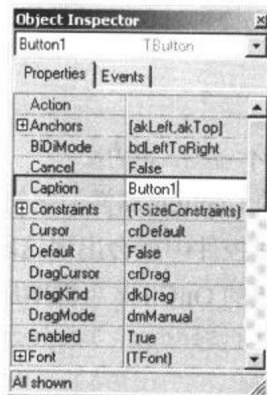


图 1-7 设置 Button1 控件的属性

(3) 用同样的方法设定其他两个控件的属性如下：

```
object Label1: TLabel //Label 控件
```

```

Left = 112 // 控件左侧距窗口边界的像素
Top = 105 // 控件上方距窗口边界的像素
Width = 3 // 宽度
Height = 13 // 高度
end
object Button2: TButton
Left = 224 // 控件左侧距窗口边界的像素
Top = 192 // 控件上方距窗口边界的像素
Width = 75 // 宽度
Height = 25 // 高度
Caption = '退出' // 按钮显示的文字
end
end

```

#### 4. 编写程序代码

(1) 选中 **Button1** 控件，选择属性窗口的 **Events** 选项卡，双击 **OnClick** 事件右侧的输入域（如图 1-8 所示），进入代码编辑窗口，同时出现程序框架。或者在窗体窗口上双击 **Button1** 控件，也会出现代码编辑窗口，如图 1-9 所示。

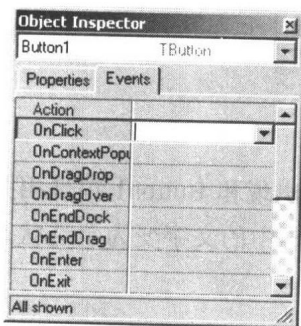


图 1-8 选择响应事件

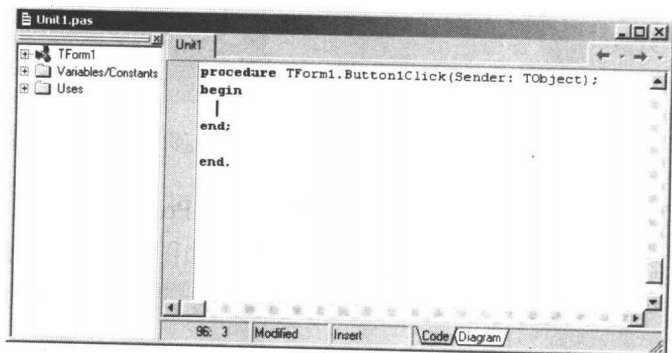


图 1-9 代码编辑窗口

(2) 在图 1-9 中输入如下代码：

```
Label1.Caption := '这是我的第一个 Delphi 程序';
```

这段程序表示设定 **Label1** 控件的 **Caption** 属性为字符串“这是我的第一个 Delphi 程序”，当用户单击“点击我”按钮时，**Label** 控件中就会出现这段文字。

(3) 用同样的方法设定退出按钮的 **OnClick** 事件响应如下：

```

procedure TForm1.Button2Click(Sender: TObject);
//退出程序
begin
Close;
end;

```

## 【程序清单】

程序完整源代码如下所示:

```
unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls;
type
  TForm1 = class(TForm)
    Label1: TLabel;
    Button1: TButton;
    Button2: TButton;
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  Form1: TForm1;
implementation
{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);
begin
  Label1.Caption:= '这是我的第一个 Delphi 程序'
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
  Close;
end;
end.
```

## 第二讲 多文档程序

### 【实现目标】

本实例将制作一个多文档程序，在一个父窗体中可以包含多个主窗体，如图 2-1 所示。使用多文档窗体便于对程序进行管理，使程序有很高的集成性。同时本程序还使用 ImageList 控件为菜单添加了图标，使用 ActionList 控件集中管理各个过程，使程序简单易读。

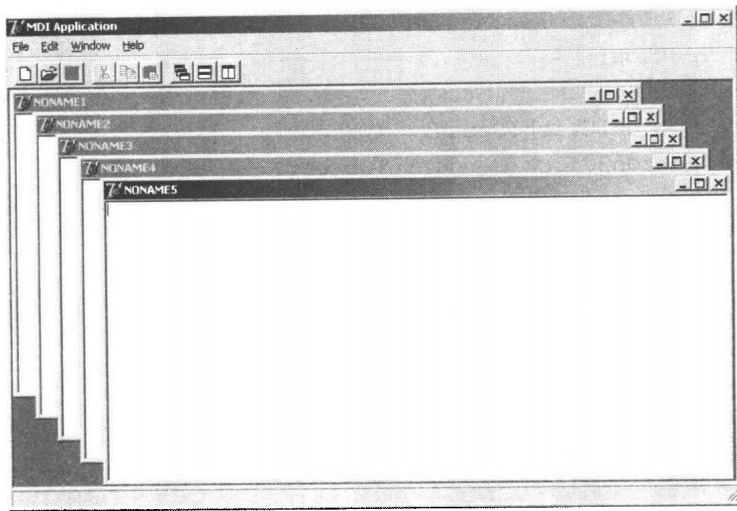


图 2-1 程序的运行效果

### 【背景知识介绍】

在应用程序中，经常会遇到这种情况：当打开一个文件时，原来的文件会被新文件覆盖，如果原文件已经修改会出现提示“是否保存修改”的对话框，即该应用程序只能打开一个窗口，比如 Windows 的“记事本”，这种应用程序在 Delphi 7.0 中统称为单文档程序 SDI (Single Document Interface)。如果在一个应用程序桌面上能同时打开多个文档窗口，并且这些窗口的外观基本一致，共用相同的系统菜单，如 Windows 的 Word 2000，这种应用程序在 Delphi 7.0 中统称为多文档程序 MDI (Multiple Document Interface)。

单文档窗口和多文档窗口的定义在 Form 对象的窗口类型 (Formstyle) 属性中设置。选择一个窗体对象，进入对象监视器，在 FormStyle 属性中有 4 个选项，各选项的含义如下：

表 2-1 FormStyle 属性各取值的含义

属性	含义
fsNormal	普通窗口，即单文档窗口，这种窗体既不是父窗体，又不是子窗体，和其他窗体的地位是平等的

续表

属性	含义
FsMDIChild	被设置成该值的窗体为多文档应用程序的子窗体，供父窗体调用
FsMDIForm	被设置成该值的窗体为多文档应用程序的父窗体，调用子窗体
FsStayOnTop	被设置成该值的窗体始终处在应用程序界面或其他窗口的最上面，直到另外一个属性也为 FsStayOnTop 的窗体覆盖它为止

## 【实现步骤及技术分析】

### 1. 新建项目

(1) 单击 File→New→Others 菜单项，将会出现 New Items 对话框，选择其中的 Projects 选项卡，如图 2-2 所示。

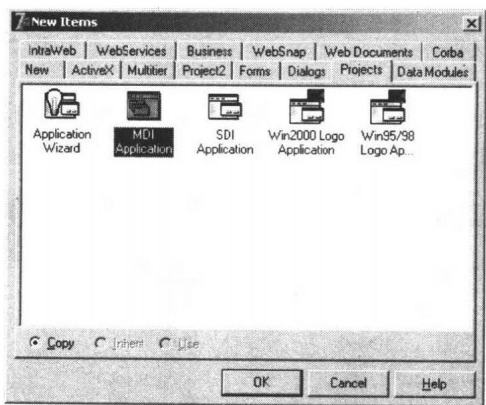


图 2-2 选择 Projects 选项卡

(2) 选择图 2-2 中的 MDI Application 项，然后单击 OK 按钮，将会出现如图 2-3 所示的选择文件保存路径对话框，这里可以将文件保存到一个合适的路径下。然后单击 OK 按钮。

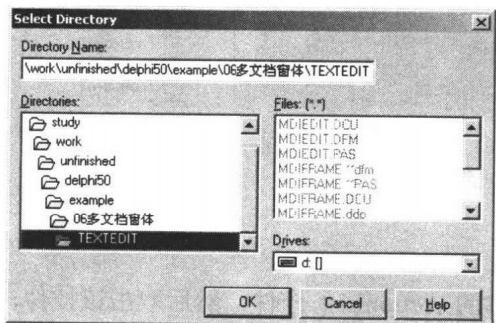


图 2-3 选择要保存的路径

(3) 此时将会出现多文档窗口的框架，如图 2-4 所示，这里显示的是程序的父窗体，可



以在对象监视器中看到该窗体的 `FormStyle` 属性被设置为 `fsMDIForm`。

(4) 单击 `View`→`Forms`，在弹出的窗体列表框中选择 `MDI Child` 窗体，此时将会出现 `MDI` 子窗体，如图 2-5 所示。在对象监视器上可以看到该窗体的 `FormStyle` 属性被设置为 `fsMDIChild`。

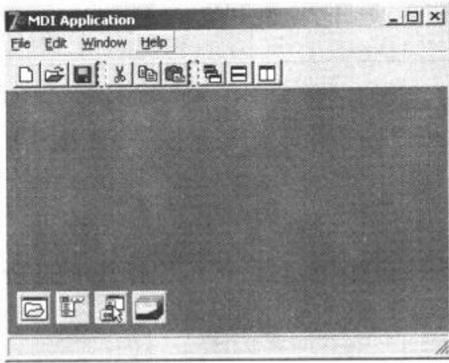


图 2-4 MDI 父窗体

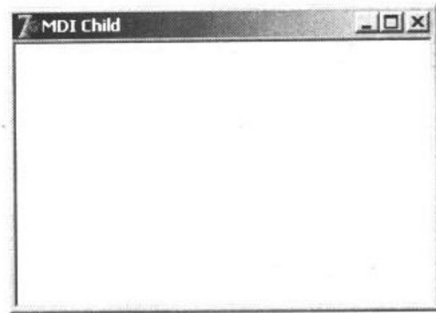


图 2-5 MDI 子窗体

(5) 切换到 `MDI` 父窗体，选择 `View`→`Object TreeView` 菜单项，将会看到向导程序自动为用户添加了多个控件，如图 2-6 所示。

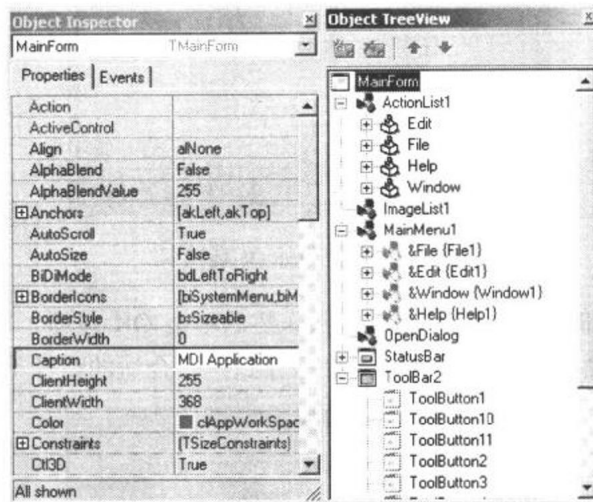


图 2-6 程序中的控件

## 2. 编写程序代码

(1) 选择 `MDI` 主窗体的 `ActionList` 控件，然后双击该控件，将会出现如图 2-7 所示的窗口，在该窗口中设定了程序要使用的各个过程的代码。双击其中的一个项目，将会切换到代码编辑器窗口，在该窗口中可以编写相应的事件处理函数。同时这里只包含了常用的新建、打开、关闭、复制、粘贴等功能，还不是很完善，读者可以在其中添加新的项目，编写需要