



Delphi

应用程序开发精彩实例

杨志国 王小琼 李世姣 编著

✚ 面向初中级读者，
从基础知识讲起，
并以实例贯穿全文

✚ 采用由浅入深的讲
解方式，结合实例
的设计思路，进行
创意与扩展

✚ 覆盖面广，涵盖
Delphi 编程的方方
面面



机械工业出版社
China Machine Press

Delphi 应用程序开发精彩实例

杨志国 王小琼 李世姣 编著



机械工业出版社

作为一个优秀的、面向对象的可视化开发工具, Delphi 拥有快捷方便的开发环境, 灵活简洁的编程语言, 以及功能强大的组件技术, 使之成为编程人员的首选编程工具。

全书共分 11 章, 列举了应用程序开发过程中多个方面的典型实例, 并分别介绍了 Delphi 在界面、菜单、工具栏、任务栏、控件应用、文件操作、数据库、报表及打印等方面的应用方法和技巧, 并随书附有本书实例的源程序。

本书作为一本 Delphi 应用程序开发技巧大全, 突出了实例典型、应用广泛等特点。

本书既可作为初学者的入门教材, 也可以作为 Delphi 开发人员宝贵的参考书籍。

图书在版编目 (CIP) 数据

Delphi 应用程序开发精彩实例/杨志国等编著.

-北京: 机械工业出版社, 2003. 1

ISBN 7-111-11613-5

I. D… II. 杨… III. Delphi 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2003) 第 007349 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

责任编辑: 郭新义 版式设计: 张丽花

北京机工印刷厂印刷·新华书店北京发行所发行

2003 年 4 月第 1 版第 1 次印刷

787mm×1092mm 1/16·26.5 印张·651 千字

0001-5000 册

定价: 39.00 元 (含 1CD)

凡购本图书, 如有缺页、倒页、脱页, 由本社发行部调换

本社购书热线电话: (010) 68993821、88379645

封面无防伪标均为盗版

前 言

作为一个优秀的、面向对象的可视化编程工具，Delphi 具有快捷方便的开发环境、灵活简洁的编程语言以及功能强大的组件技术，逐渐成为编程人员的首选编程工具，特别对于一个熟悉其他编程工具的人员来说，感受会更加深刻。无怪乎有人说：聪明的程序员使用 Delphi。

面向对象的程序开发已经成为目前软件开发的一种潮流。现代计算机技术和信息技术的高速发展，必然会对应用软件的开发、应用和普及速度提出更高的要求。如何利用一种可靠、高效、快捷、功能强大的开发工具来完成应用程序的开发，已成为程序开发人员必须考虑的问题。Delphi 作为一个全新的开发工具，其面向对象、组件技术、集成环境的优势都决定了它是一种优秀的且可以信赖的程序开发工具。

本书以实例典型、应用广泛为特点，列举了 Delphi 在应用程序开发中不同方面的 100 个实例。这些实例基本上概括了应用程序开发的各个方面，均具有很高的应用价值。书中并不只是对技巧的简单介绍，而是紧密结合了 Windows 应用程序的开发特点，介绍了各个实例的应用原理，突出实例的技术要点，而且对实例进行了详细分析，使读者能够举一反三。本书中实例的源程序代码均可在配书光盘中找到。

本书作为一本 Delphi 应用程序开发的技巧大全，既可作为初学者的入门教材，也可以作为 Delphi 开发人员宝贵的参考书籍。

编者

2003 年 1 月

目 录

第1章 界面	1
实例1 制作闪现窗口	2
实例2 MDI 窗口客户区显示图片	6
实例3 不规则窗口设计	10
实例4 无边界窗口的移动	13
实例5 限制窗口大小	15
实例6 自适应窗口设计	17
实例7 窗口融合技术	20
实例8 窗口背景设置	22
第2章 菜单、工具栏、任务栏	25
实例9 能任意拖动的工具栏	26
实例10 MDI 窗口菜单融合技术	28
实例11 定制窗口系统菜单	30
实例12 任务栏应用程序设计	34
实例13 动态创建菜单	39
第3章 控件应用	43
实例14 控件的动态创建及应用	44
实例15 使用 TreeView 维护数据库	51
实例16 控件标题竖排显示	55
实例17 定制 TDBGrid	60
实例18 定制 TMemo	66
实例19 定制 TRichEdit	69
实例20 控件背影制作	74
第4章 文件操作	79
实例21 文本文件操作	80
实例22 INI 文件操作	83

实例 23 资源文件的操作.....	89
实例 24 Excel 文件操作.....	91
实例 25 Word 文件操作.....	96
实例 26 类型文件操作.....	100
第5章 数据库	107
实例 27 在程序中维护数据库别名.....	108
实例 28 瘦型数据库设计.....	113
实例 29 动态生成数据库表格.....	117
实例 30 使用 TQuery 维护数据库.....	122
实例 31 获取数据库结构.....	126
实例 32 使用 ADO 控件存取数据库.....	130
实例 33 ADO 的 BatchUpdate 方法的操作.....	134
实例 34 动态设置数据源.....	138
实例 35 数据字典维护.....	141
实例 36 定制 SQL 复杂查询.....	145
实例 37 定制数据录入窗口.....	151
实例 38 TBatchMove 实例数据批处理.....	157
实例 39 应用程序自动登录数据库.....	161
实例 40 多表联合查询.....	164
实例 41 Word 文件在数据库的存取.....	168
第6章 报表及打印	173
实例 42 更改默认的打印机.....	174
实例 43 打印中更改纸张大小.....	177
实例 44 获取打印机队列.....	181
实例 45 直接向打印机输出一行.....	185
实例 46 打印文本编辑控件内容.....	188
实例 47 图片打印.....	191
实例 48 打印 TDBGrid 内容.....	193
实例 49 套打程序设计.....	200
实例 50 动态生成报表.....	203
实例 51 通过 Excel 生成报表.....	207
实例 52 定制 Qreport 预览窗口.....	211
实例 53 制作多栏报表.....	216

第7章 图形、图像及多媒体	223
实例 54 从文件中提取图标	224
实例 55 将图形文件放入 EXE 文件中	227
实例 56 抓屏程序	231
实例 57 墙纸更换程序	235
实例 58 媒体播放器设计	240
实例 59 制作屏保程序	245
实例 60 动画设计	249
实例 61 英语朗读设计	252
实例 62 图形显示特技	256
实例 63 滚动字幕	263
实例 64 播放 Flash 文件	267
实例 65 光驱的打开与关闭	271
实例 66 光驱的自动运行设置	273
实例 67 空心字与透明字	276
实例 68 图形镜像应用	279
第8章 网络	285
实例 69 获得本机 IP 地址和网卡地址	286
实例 70 邮件发送程序	288
实例 71 在程序中实现超链接	293
实例 72 监控程序设计	296
实例 73 聊天程序设计	301
第9章 系统及程序控制	307
实例 74 获取系统信息	308
实例 75 程序自启动和退出关机	312
实例 76 进程管理程序	316
实例 77 屏蔽系统按键	322
实例 78 防止程序运行多个实例	325
实例 79 控制面板程序	328
实例 80 鼠标拖动控件	331
实例 81 动态链接库使用	336
实例 82 定义应用程序热键	340
实例 83 程序间数据传递	343

实例 84	键盘及鼠标操作回放.....	347
实例 85	动态数据交换技术 (DDE) 应用.....	351
实例 86	在应用程序中操作其他程序.....	355
实例 87	显示工作组中的计算机.....	358
实例 88	获得光标位置窗口或文本.....	362
实例 89	以回车键代替 Tab 键.....	364
实例 90	显示键盘状态键.....	366
实例 91	利用注册表保存程序信息.....	369
第 10 章	异常处理和提示信息.....	375
实例 92	提示框和对话框的中文显示.....	376
实例 93	定制消息框.....	379
实例 94	在消息框中输出表格.....	381
第 11 章	安全管理.....	383
实例 95	为应用程序加入密码功能.....	384
实例 96	共享软件的限次设计.....	388
实例 97	实现软件注册功能.....	391
实例 98	通过程序管理 IE 配置.....	397
实例 99	应用程序加入菜单权限管理.....	404
实例 100	为应用程序制作安装盘.....	408

第1章 界面

本章概要

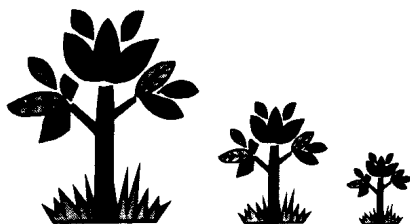
主要介绍 Delphi 应用程序开发过程中关于界面设计的技巧和方法，同时介绍 Delphi 应用程序界面设计中的一些特殊要求及其实现方法。

界面设计是 Delphi 应用程序开发过程的基础部分，也是 Windows 应用程序开发过程中最重要的组成部分。应用程序的界面友好性主要体现在界面设计的应用技巧上。

本章实例

- 制作闪现窗口
- MDI 窗口客户区显示图片
- 不规则窗口设计
- 无边界窗口的移动
- 限制窗口大小
- 自适应窗口设计
- 窗口融合技术
- 窗口背景设置

通过这些有关界面设计的实例，将会展示 Delphi 在开发应用程序界面设计方面的强大功能和优势。



实例 1 制作闪现窗口

通常，应用程序总会在启动时出现一个窗口，用于显示应用程序的版权、版本，甚至是一些欢迎词之类的信息。因为这个窗口出现在应用程序之前，显示几秒钟后即自动关闭，所以称之为闪现窗口，也可以说是软件封面。本实例的源程序见光盘目录 S01。

设计原理：

制作闪现窗口的方法：针对个别应用程序初始化时间较长的特点，在应用程序进行初始化时显示闪现窗口，直到应用程序初始化完成之后自动关闭它。

技术要点：

方法一：采用计时器 Timer，延迟闪现窗口的出现时间，在一段时间后自动关闭。

方法二：采用函数配合应用程序的初始化过程，完成闪现窗口的延迟处理。

实例分析：

（一）手工或以计时器的方式关闭窗口

1. 新建项目“Project1”。
2. 以 Form1 作为闪现窗口。设置 Form1 的属性 BorderStyle 值为 bsNone，属性 Position 值为 poScreenCenter。
3. 在 Form1 中设置如图 1-1 所示的版面信息。
4. 在 Form1 中放置计时器“Timer1”，设置其属性 Interval 值为 3000，实现闪现窗口出现 3 秒钟后自动关闭。
5. Form1 的界面设计如图 1-1 所示。



图 1-1 Form1 的界面设计

6. 新建 Form2 作为主窗口。选择 Delphi 主菜单 Project 中的 Option 项，在“Main Form”下拉式列表框中将程序的主窗口设置为 Form2。

7. 在 Form1 的 OnClick 事件中加入以下代码:

```
procedure TForm1.FormClick(Sender: TObject);
begin
  Close; //在 Form1 未自动关闭之前, 可以单击它手动关闭, 同时将 Image1 的 OnClick 事件指向 FormClick
end;
```

8. 在控件 Timer1 的 OnTimer 事件中加入以下代码:

```
procedure TForm1.Timer1Timer(Sender: TObject);
begin
  Close; //通过计时器使 Form1 出现一定时间后自动关闭
end;
```

9. 在 Form2 中“关于”菜单项的 OnClick 事件中加入以下代码:

```
procedure TForm2.N1Click(Sender: TObject);
begin
  Form1.ShowModal; //通过菜单显示 Form1
end;
```

10. 在 Form2 的 OnActive 事件中加入以下代码:

```
procedure TForm2.FormActivate(Sender: TObject);
begin
  Form1.ShowModal; //在 Form2 出现之前显示 Form1
end;
```

11. 编译程序, 即可查看 Form1 的显示效果。

(二) 利用函数配合应用程序初始化过程, 完成闪现窗口的自动关闭

1. 新建项目“Project1”。

2. 以 Form1 作为闪现窗口, 设置其属性 BorderStyle 值为 bsNone, 属性 Position 值为 poScreenCenter, 并在 Form1 中设置其版面信息。版面设计与图 1-1 相似, 只是不需要使用计时器控件。

3. 新建一个 Form2 作为主窗口, 通过选择 Delphi 主菜单 Project 中的 Option 项, 将 Main Form 设置为“Form2”。

4. 通过选择 Delphi 主菜单 Project 中的 Option 项, 将 Form1 从 Auto-Create 中移到 Available Forms 中。

5. 项目文件 Project1 代码如下:

```

program Project1;
uses
  Forms, Windows, //在此处增加 Windows
  Unit1 in 'Unit1.pas' {Form1},
  Unit2 in 'Unit2.pas' {Form2};

{$R *.res}
var ltime:TDateTime;
begin
  Form1:=TForm1.Create(Application); //在程序中创建闪现窗口 Form1
  Form1.Show;
  Form1.Update;
  Application.Initialize;
  Application.CreateForm(TForm2,Form2);
  lTime:=GetTickCount;
  While((GetTickCount-lTime)/1000<3) do; //延迟 3 秒
  Form1.Free; //释放
  Application.Run;
end.

```

举一反三:

以上两种设计闪现窗口的方法都是基于一般窗口模式的。在使用 MDI 窗口时,可以将闪现窗口显示到主窗口正中偏上的位置,填补主窗口中客户区背景的空白区域。其实现方法如下:

1. 新建项目“Project1”。
2. 以 Form1 作为闪现窗口,设置其属性 BorderStyle 值为 bsNone,属性 FormStyle 值为 fsStayOnTop,属性 Position 值为 poScreenCenter,并在 Form1 中设置其版面信息。
3. Form1 的界面设计如图 1-2 所示。

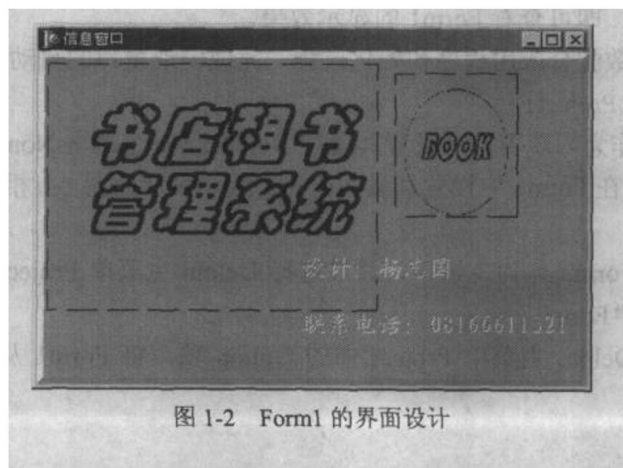


图 1-2 Form1 的界面设计

4. 新建 Form2 作为主窗口, 设置其属性 FormStyle 值为 fsMDIForm, 属性 Position 值为 poScreenCenter, 并进行其他方面的设置。Form2 的界面设计如图 1-3 所示。

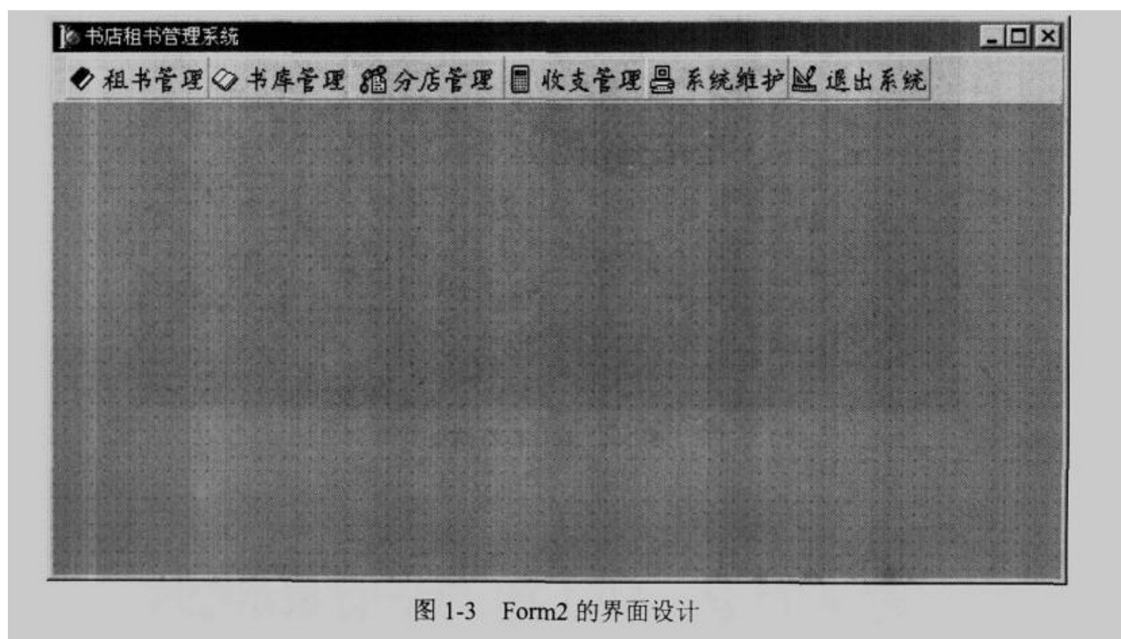


图 1-3 Form2 的界面设计

5. 项目文件代码如下:

```
program Project1;
..... (此处代码见光盘)
{$R *.res}
begin
    Form1:=TForm1.Create(Application); //信息窗口
    Form1.Show;
    Form1.Update;
    Application.Initialize;
    Form1.Panell1.BevelOuter:=bvNone; //不显示边框线
    Application.CreateForm(TForm2,Form2);
    Application.Run;
end.
```

6. 程序运行结果如图 1-4 所示。

这种方法只是将闪现窗口作为顶层窗口居中的形式显示到 MDI 主窗口的中间, 填补了主窗口的空白。如果要在主窗口中启动其他窗口, 必须在其他窗口显示之前用 hide 方法将窗口 Form1 隐藏, 以免影响其他 MDI 子窗口的显示。另外, 为了更好地解决 MDI 客户区的信息显示问题, 在后面的小节中还会介绍如何在 MDI 的客户区中放入图片。



图 1-4 程序运行结果

实例 2 MDI 窗口客户区显示图片

在多文档窗口（MDI）中，主窗口中的客户区作为一个独立的窗口，用于显示子窗口。在一般情况下，这种窗口是空白的，没有太大的作用。且在正常情况下，不能在它的上面放入图片，因为在加入图片之后，图片将会遮挡住子窗口的显示。但如果设法在主窗口的客户区中放上如公司徽标之类的信息或图片，将使整个界面更加紧凑、美观。本实例源程序见光盘目录 S02。

设计原理：

通过调用操纵窗口客户区的 Windows API 函数，捕获窗口消息 WM_ERASEBKND，并改变窗口默认的动作，从而实现自定义的图片输出。

技术要点：

使用窗口方法 ClientWndProc()，捕获窗口消息 WM_ERASEBKND，发送信息到当前窗口，并对窗口中的图片进行重画，从而实现在 MDI 客户区的图片输出。

实例分析：

1. 新建项目“Project1”。
2. 设置 Form1 的 FormStyle 属性值为 fsMDIForm。
3. 在窗口中放入图片控件“Image1”。将控件 Image1 的 Picture 的属性值设置为 TBmp；将控件 Image1 的 Stretch 属性值设置为 True。
4. Form1 的界面设计如图 1-5 所示。

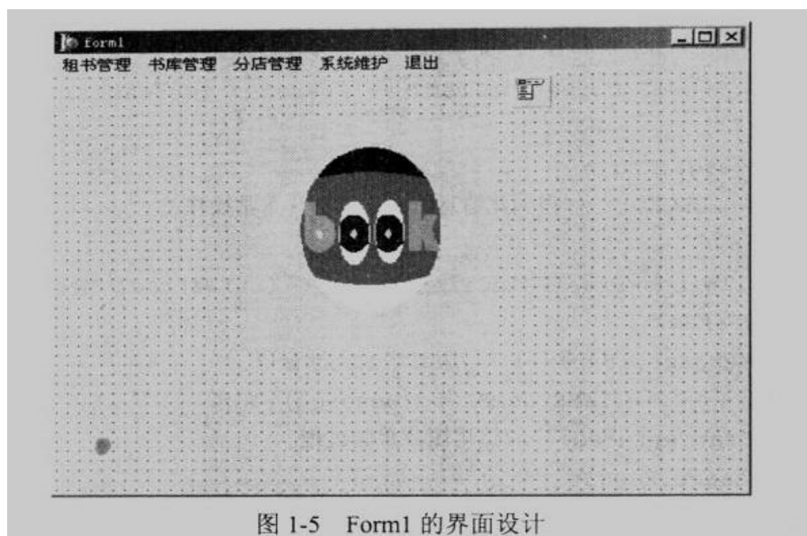


图 1-5 Form1 的界面设计

5. Form1 单元文件代码如下:

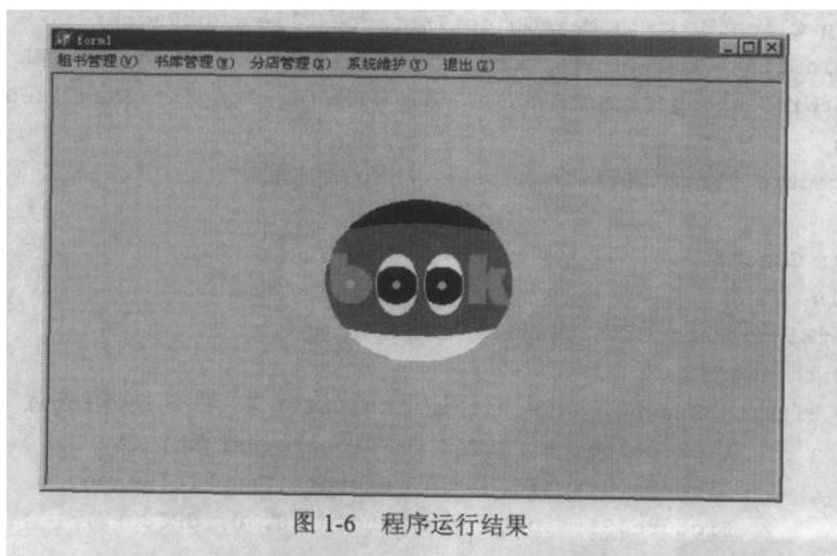
```
unit Unit1;
..... (此处代码见光盘)
protected
    procedure CreateWnd; override;
end;
var
    Form1: TForm;
implementation
{$R *.DFM}
procedure TForm1.CreateWnd;0
begin
    Inherited CreateWnd;
    FNewClientProc := MakeObjectInstance(ClientWndProc);
    FOldClientProc := Pointer(GetWindowLong(ClientHandle, GWL_WNDPROC));
    SetWindowLong(ClientHandle, GWL_WNDPROC, LongInt(FNewClientProc));
end;
procedure TForm1.DrawCentered; //图片居中重画
var
    CR: TRect;
begin
    GetWindowRect(ClientHandle, CR);
    with image1 do
        BitBlt(FDrawDC, ((CR.Right - CR.Left) - Picture.Width) div 2,
            ((CR.Bottom - CR.Top) - Picture.Height) div 2,
            Picture.Graphic.Width, Picture.Graphic.Height,
            Picture.Bitmap.Canvas.Handle, 0, 0, SRCCOPY);
```

```

end;
procedure TForm1.ClientWndProc(var Message: TMessage);
begin
    case Message.Msg of
        WM_ERASEBKGND: //得到此消息时, 处理图片重画操作
            begin
                CallWindowProc(FoldClientProc, ClientHandle, Message.Msg,
                    Message.wParam,
                    Message.lParam); //发送消息到原窗口
                FDrawDC := TWMEraseBkGnd(Message).DC;
                DrawCentered; //调用图片重画过程
                Message.Result := 1;
            end;
        WM_VSCROLL, WM_HSCROLL: //得到这两个消息时, 保证图片及时刷新
            begin
                Message.Result := CallWindowProc(FoldClientProc, ClientHandle,
                    Message.Msg,
                    Message.wParam, Message.lParam);
                InvalidateRect(ClientHandle, nil, True);
            end;
        else
            Message.Result := CallWindowProc(FoldClientProc, ClientHandle, Message.Msg,
                Message.wParam, Message.lParam);
        end;
    end;
end.

```

6. 程序运行结果如图 1-6 所示。



举一反三：

上述方法实现了在 MDI 窗口上放置图片，它将一副标志放在 MDI 客户区的正中。只要对上例略加修改，即可实现图片在客户区的平铺与伸展，从而实现不同的应用效果。

1. 图片拉伸的实现代码如下：

```
procedure TForm1.DrawStretched;  
var  
    CR: TRect;  
begin  
    .....(此处代码见光盘)  
end;
```

2. 图片平铺的实现代码如下：

```
procedure TForm1.DrawTiled;  
var  
    Row, Col: Integer;  
    CR, IR: TRect;  
    NumRows, NumCols: Integer;  
begin  
    .....(此处代码见光盘)  
end;
```

3. 图片拉伸与平铺的结果如图 1-7、图 1-8 所示。

