

1000  
万水创作效果百例丛书

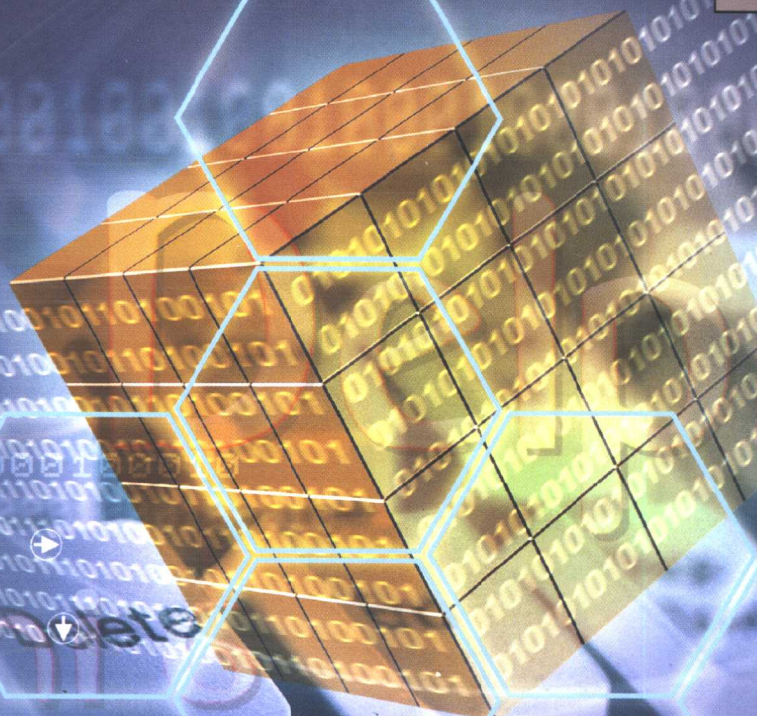
# Delphi 6

## 精彩编程百例

张龙卿 尹建民 等编著



附光盘  
CD-ROM



99  
Programming  
中国水利水电出版社  
www.waterpub.com.cn

万水创作效果百例丛书

# Delphi 6 精彩编程百例

张龙卿 尹建民 等编著

中国水利水电出版社

## 内 容 提 要

本书由浅入深、循序渐进，以实例分析的形式，全面介绍了 Delphi 6 各方面的功能及开发不同类型应用程序的知识，全书的 100 个实例被划分为 9 部分，包括基本组件与基础知识、图形图像及多媒体、使用对话框、建立数据报表、建立数据库应用程序、字符串处理、文件的处理、建立 Internet 应用程序、高级应用及其他等方面。所有实例的讲解都包括实例说明、核心要点、实现步骤、归纳注释及源代码等内容。不但讲述了实例的内容，还讲述了核心技术、技巧，并分析了关键代码及实现步骤。本书与其他类似书籍相比的最大特点就是：每个实例都非常具有代表性，也非常实用。

本书既适合于初学者、大专院校的学生使用，也适合于初中级程序员使用，还可以作为各种培训班提高学生 Delphi 水平的上机教程。

### 图书在版编目 (CIP) 数据

Delphi 6 精彩编程百例/张龙脚等编著. —北京: 中国水利水电出版社, 2002  
(万水创作效果百例丛书)

ISBN 7-5084-1113-7

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

中国版本图书馆 CIP 数据核字 (2002) 第 036588 号

书 名	Delphi 6 精彩编程百例
作 者	张龙脚 尹建民 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: <a href="http://www.waterpub.com.cn">www.waterpub.com.cn</a> E-mail: <a href="mailto:mchannel@public3.bta.net.cn">mchannel@public3.bta.net.cn</a> (万水) <a href="mailto:sale@waterpub.com.cn">sale@waterpub.com.cn</a>
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	北京蓝空印刷厂
规 格	787×1092 毫米 16 开本 21.25 印张 466 千字
版 次	2002 年 6 月第一版 2002 年 6 月北京第一次印刷
印 数	0001—5000 册
定 价	34.00 元 (含 1CD)

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

Delphi 由于其使用简单、学习容易及功能强大等特点，正越来越受到开发人员的欢迎。特别是 Delphi 6 的发布，使其他程序开发工具受到了前所未有的冲击。Delphi 6 在实现开发功能强大的应用程序的同时，在开发 Internet 应用程序、数据库应用程序及跨平台的应用程序等方面，实现了新的飞越，已经充分展现出比其他开发工具更大的优越性。业界对 Delphi 有这样的评价：真正的程序员使用 C++，聪明的程序员使用 Delphi。使用 Delphi 开发应用程序，可以达到事半功倍的效果。

尽管目前书店中讲解 Delphi 的书籍很多，讲解实例的书籍也有一些，但真正做到知识覆盖面广且能将每个实例剖析透彻的并不多见。正是从这一方面考虑，我们几位使用 Delphi 开发项目多年的程序员，深感有必要将我们平时积累的丰富编程经验及大量的 Delphi 典型实例与大家进行交流，如果大家的编程水平能从根本上得到提高，将是我们最大的欣慰。

本书讲解了 100 个实例，并将其划分为 9 部分，包括基本组件与基础知识、图形图像及多媒体、使用对话框、建立数据报表、建立数据库应用程序、字符串处理、文件的处理、建立 Internet 应用程序、高级应用及其他等方面。所有实例的讲解都包括实例说明、核心要点、实现步骤、归纳注释及源代码等内容。不但讲述了实例的内容，还讲述了核心技术、技巧，并分析了关键代码及实现步骤。本书与其他类似书籍相比的最大特点就是：每个实例都非常具有代表性，都是从实际应用中抽象出来的。

本书由多人合作完成，主要由张龙卿、尹建民、刘伟、吴新平、王涛、马新利、孙爱琴、吴齐等同志编写，另外，苏霞、董威、郭清明、楼鸣、夏遂冬、张宁丽、秦悦、王凯、罗粹、董蕾、章钢、张挺、隋庆、周微奥、崔祥、于强、董丰、孙知、张仑、隋钎、杜达、王芹、姜辉、武直冰、辛翔、郑乐、俊清、宿令星、丁其、王新利、张和平、范军、董国平、隋盟、郭芝、夏海东、杜莘等同志也参加了部分内容的编写及素材整理工作。在此一并表示感谢。

由于时间仓促、作者水平有限，错误及不当之处在所难免，敬请广大读者批评指正。

编者

2002 年 3 月

# 目 录

前言

## 第一部分 基本组件与基础知识

实例 1	简单程序 .....	2
实例 2	使用 ListBox .....	4
实例 3	建立主菜单和弹出菜单 .....	6
实例 4	多文本编辑器 .....	13
实例 5	使用 TRadioGroup 和 TScrollBar .....	15
实例 6	使用 TScrollBar 与 TUpDown .....	18
实例 7	单选按钮与复选按钮 .....	26
实例 8	使用 TStringGrid 组件 .....	29
实例 9	动态创建 TPageControl 标签页 .....	32
实例 10	使用滚动条控制多标签页 .....	36
实例 11	建立计时器 .....	39
实例 12	使用 TSpinEdit 和 TSpinButton 组件 .....	42
实例 13	浏览日历 .....	48
实例 14	字体的自动伸缩 .....	51
实例 15	建立组件库 .....	53
实例 16	使用 TActionManager 相关组件 .....	55
实例 17	使用 OLE 嵌入其他对象 .....	59
实例 18	使用鼠标光标 .....	62
实例 19	处理键盘触发事件 .....	65
实例 20	响应鼠标事件 .....	67
实例 21	使用 THotKey 组件 .....	70
实例 22	循环修改字体颜色 .....	73
实例 23	随机产生颜色及传递颜色 .....	76
实例 24	查找文件 .....	80

## 第二部分 图形、图像及多媒体

实例 25	显示位图、图标及图元 .....	85
-------	------------------	----

实例 26	绘制图形 .....	88
实例 27	拷贝屏幕 .....	91
实例 28	组合调色板 .....	94
实例 29	显示并控制图像 .....	98
实例 30	使用 TPaintBox 组件绘图 .....	104

### 第三部分 使用对话框

实例 31	使用输入对话框 .....	107
实例 32	使用对话框进行猜数游戏 .....	110
实例 33	字体对话框与颜色对话框 .....	114
实例 34	打开及保存文件对话框 .....	116
实例 35	查找与替换对话框 .....	119

### 第四部分 建立数据报表

实例 36	建立简单数据报表 .....	124
实例 37	分组报表 .....	129
实例 38	图像报表 .....	133
实例 39	复合报表 .....	137
实例 40	合成多个报表 .....	142

### 第五部分 建立数据库应用程序

实例 41	配置 BDE .....	146
实例 42	使用 TTable 开发简单程序 .....	148
实例 43	使用 TQuery 组件 .....	150
实例 44	使用 TTable 的 Filter 属性 .....	152
实例 45	使用 TStoredProc 组件操作数据 .....	155
实例 46	使用 TDBImage 组件显示图像数据 .....	158
实例 47	使用 TBDEClientDataSet 组件 .....	160
实例 48	建立 ADO 数据库连接 .....	162
实例 49	用 TADOQuery 查询两基表数据 .....	165
实例 50	用 TADOTable 建主从表程序 .....	168
实例 51	使用 TADODataset 组件 .....	171
实例 52	使用 TADOCommand 组件 .....	175
实例 53	在 TADOQuery 中使用参数 .....	178
实例 54	使用 TIBTable 组件 .....	181

实例 55	TIBDatabaseInfo 显示数据库信息 .....	183
实例 56	TClientDataSet 和 TDataSetProvider .....	186
实例 57	使用 TSQLClientDataSet 组件 .....	189
实例 58	使用 TDBGrid 组件 .....	191
实例 59	DBComboBox 和 DBListBox .....	195
实例 60	TDBCheckBox 和 TDBRadioGroup .....	197
实例 61	使用 TDBLookupListBox 组件 .....	200
实例 62	用 TDBCtrlGrid 组件显示数据 .....	202
实例 63	组合及模糊查询的实现 .....	204
实例 64	使用 TDBChart 显示直方图 .....	210
实例 65	应用 Decision Cube 面板组件 .....	212
实例 66	使用 TBatchMove 移动批记录 .....	214
实例 67	使用 TSession 显示数据库别名 .....	217

## 第六部分 字符串处理

实例 68	拷贝字符串 .....	220
实例 69	插入或删除字符串 .....	223
实例 70	使用 Pos 方法查找字符串 .....	227

## 第七部分 文件的处理

实例 71	读取文本文件内容 .....	231
实例 72	选择文件列表 .....	233
实例 73	使用 FileSearch 方法查找文件 .....	235
实例 74	用 Append 方法追加文件内容 .....	237
实例 75	用 FilePos 方法获得文件位置 .....	240
实例 76	用 ChangeFileExt 修改文件扩展名 .....	243
实例 77	使用 DiskSize 和 DiskFree 方法 .....	246

## 第八部分 建立 Internet 应用程序

实例 78	获得计算机串行口 .....	249
实例 79	显示计算机联网信息 .....	251
实例 80	显示 Modem 的状态 .....	253
实例 81	检测网络的连接状态 .....	256
实例 82	用 TWebBrowser 设计浏览器 .....	258
实例 83	使用 TPowersock 组件 .....	266

实例 84	使用 TNMMsg 和 TNMMSGServ .....	269
实例 85	使用 TNMHTTP 组件 .....	272
实例 86	使用 TNMUDP 组件 .....	275
实例 87	使用 TNMUUEProcessor 组件 .....	279
实例 88	使用 TNMURL 组件 .....	283
实例 89	使用 TNMFinger 组件.....	285
实例 90	使用 TNMPOP3 组件.....	288
实例 91	使用 TNMSMTP 组件.....	295

### 第九部分 高级应用及其他

实例 92	建立透明表单 .....	305
实例 93	隐藏窗口标题 .....	307
实例 94	显示粘贴板上的图像 .....	309
实例 95	获得计算机的显卡类型 .....	311
实例 96	TWordApplication 和 TWordDocument .....	313
实例 97	显示系统字体 .....	317
实例 98	读写注册表数据 .....	320
实例 99	建立资源管理器 .....	324
实例 100	使用 TListView 等管理文件 .....	327



## 第一部分

### 基本组件与基础知识

该部分实例主要讲解一些基本控件、基本功能。开发 Delphi 应用程序时，这些基本功能的使用最频繁，掌握这些内容是开发应用程序的基础。通过分析这些实例，可以掌握使用 Delphi 6 设计应用程序的基本方法与技巧。

**Let's GO!**



## 实例

## 1

## 简单程序



## 实例说明

该实例建立一个最简单的 Delphi 应用程序，界面上只有一个标签和一个按钮，但是该实例包含了设计 Delphi 应用程序的基本步骤及基本结构。

该实例的运行结果如图 1-1 所示。



图 1-1 应用程序的执行结果



## 操作步骤

- ① 使用 File|New|Application 菜单新建一个应用程序。
- ② 选择 Standard 组件面板的标签和按钮组件，在表单中分别插入一个标签和按钮。
- ③ 将标签的 Caption 属性设置为“欢迎光临!”，并将字体设置为“楷体蓝色|加粗|2 号”。
- ④ 将按钮的 Caption 属性修改为“退出”。
- ⑤ 打开对象观察器的 Events 面板，为按钮添加一个 OnClick 事件，然后在该事件管理器中添加关闭应用程序的代码。
- ⑥ 如果没有添加退出按钮，可以通过单击表单右上角的退出“×”按钮退出应用程序。



## 技术要点

- 该实例通过应用程序对象 Application 的 Terminate 方法关闭应用程序。如果要在代码中使用一个对象或组件的方法或属性等，Delphi 提供了很好的联机帮助，只要在代码中输入该对象或组件的名字，后面再输入一个小点后，很快就会显示出该对象具有的属性及方法的列表，可以从该列表中选择需要的属性或方法。

- 当然，如果对一些对象及方法比较熟悉，可以直接输入这些对象的属性或方法。



## 归纳注释

建立一个 Delphi 应用程序一般都需要经过添加组件、设置属性及事件、添加执行代码等步骤。

该应用程序的关键源代码如下:

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls;

type
  TForm1 = class(TForm)
    Label1: TLabel;
    Button1: TButton;
    procedure Button1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject);    //单击“退出”按钮
begin
  Application.Terminate;    //终止应用程序
end;

end.
```



## 实例说明

该实例说明了 ListBox 的主要作用、关键属性及方法。在界面中设置了一个编辑框，当在编辑框中输入内容并单击按钮后，则会将该内容添加到列表框中；如果在列表框中某一项上双击，则会显示出当前选择项的序列号。在列表框中添加几项后，在第 2 项上双击，则该应用程序执行的结果如图 2-1 所示。

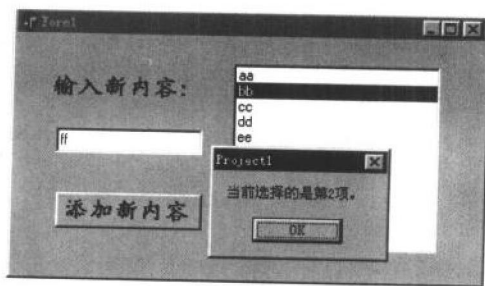


图 2-1 应用程序的执行结果



## 操作步骤

- ① 使用 FileNewApplication 菜单新建一个应用程序。
- ② 在表单中添加一个标签 Label1、一个编辑框 Edit1、一个按钮 Button1 和一个列表框 ListBox1。
- ③ 将 Label1 的 Caption 属性修改为“输入新内容:”；将 Button1 的 Caption 属性修改为“添加新内容”。
- ④ 为按钮添加 OnClick 事件，用于向列表框中添加新的项。
- ⑤ 为列表框添加 OnDblClick 事件，用于显示当前选择的一个项的位置。



## 技术要点

- 要向列表框中添加内容，则需要使用 ListBox 组件的 Add 方法。
- 通过列表框的 ItemIndex 属性获得一个项的索引值。由于该属性从 0 开始计算，所以一个项在列表框中的顺序需要在该属性值上加 1。
- 要显示一个对话框提示信息，可以使用 ShowMessage(string)方法，其参数是一个字符串，用来表示显示的信息。



## 归纳注释

列表框是使用比较频繁的组件，可以很方便地增加或删除一些项，非常适合于多个项的列

表管理。

该应用程序的源代码如下：

```
unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls;
type
  TForm1 = class(TForm)
    ListBox1: TListBox;
    Edit1: TEdit;
    Button1: TButton;
    Label1: TLabel;
    procedure Button1Click(Sender: TObject);
    procedure ListBox1DbClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
var
  Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.Button1Click(Sender: TObject); //单击“添加新内容”按钮
begin
  if Edit1.Text='' then
    ShowMessage('请在编辑框中输入新内容。')
  else
    ListBox1.Items.Add(Edit1.Text); //向列表框中添加新项
end;

procedure TForm1.ListBox1DbClick(Sender: TObject); //双击列表框
begin
  ShowMessage('当前选择的是第'+IntToStr(ListBox1.ItemIndex+1)+'项。')
end;
end.
```



## 实例说明

Delphi 提供了一个非常完整的使用菜单的例子，位于...\\help\\samples\\menu 目录下。本实例对该例子进行了修改，并将通过分析该实例说明主菜单和弹出菜单的设计方法。

系统主菜单需要通过 Standard 组件面板上的 MainMenu 组件建立，弹出菜单则通过该组件面板上的 PopupMenu 组件实现。在应用程序编辑状态下，可以在这两个组件上面双击来打开菜单编辑器编辑新的菜单。

本实例主要通过主菜单和弹出菜单改变图形的形状、颜色等。运行结果如图 3-1 所示。

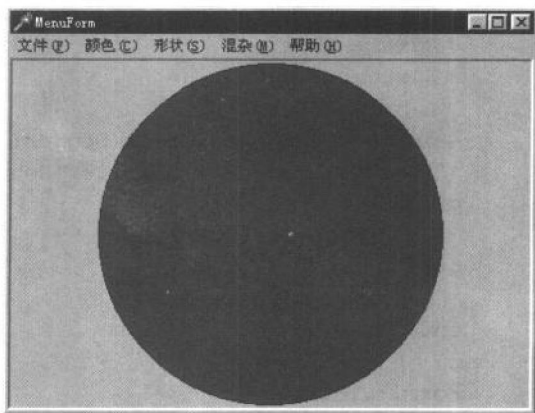


图 3-1 应用程序的执行结果



## 操作步骤

- ① 新建一个应用程序，将表单的名字修改为 MenuForm。
- ② 在表单中先放置一个面板，将其名字修改为 DisplayPanel，其 Align 属性设置为 alClient，这样面板会使用表单的整个客户区域。再在表单中放置一个主菜单 MainMenu、一个弹出菜单 PopupMenu、两个颜色对话框 SolidColorDialog 和 AnyColorDialog、一个图形 DemoShape。
- ③ 双击主菜单 MainMenu，并添加菜单项及菜单，然后为各个菜单设置 OnClick 事件并添加合适的代码。按照同样的方法，为 PopupMenu 添加 3 个弹出菜单，并添加 OnClick 事件及合适的代码。最后将 MenuForm 的 Menu 属性设置为 MainMenu。
- ④ 将 DemoShape 的 Shape 属性设置为 stCircle，即显示一个圆形。
- ⑤ 选择面板 DisplayPanel，将其 PopupMenu 属性设置为已经定义的弹出菜单 PopupMenu，这样在整个面板区域内的任意位置都可以弹出该菜单。

打开该面板对应的对象观察器的 Events 面板，在 OnResize 中双击，则会添加一个在面板的大小改变时触发的事件，通过该事件设置图形 DemoShape 的大小。其大小将根据面板的变化而变化。

- ⑥ 通过 File|New|Form 菜单新建第二个表单，并将其名字修改为 AboutBox，在该表单中

添加两个标签及一个“退出”按钮。

⑦ 由于 MainMenu 中的“帮助|关于”菜单可以打开 AboutBox 表单，而它们又不在同一个单元文件中，所以主表单需要通过 File|Use Unit 菜单调用 AboutBox 表单的单元文件。

最终的设计界面如图 3-2 所示。

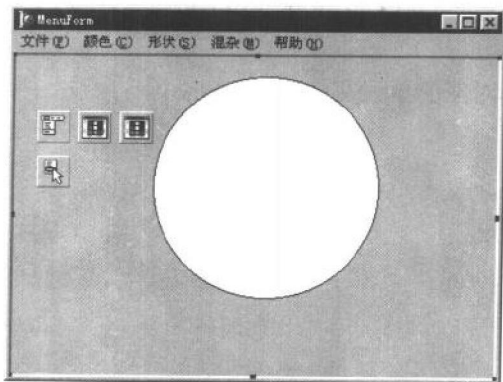


图 3-2 最终完成的设计界面



### 技术要点

- 设计菜单时，应在菜单组件上面双击，然后选择一个虚线框并打开对象观察器，在 Name 属性中为菜单输入一个名字，在 Caption 属性中输入菜单显示的标题。如果要为菜单设置一个快捷键，可以在一个字母前添加“&”符号，如文件使用的快捷菜单是&F。
- 如果要插入新的菜单，可以在一个菜单上面右击，从弹出的快捷菜单中选择 Insert，则会插入一个虚线框。主菜单项会插入在选择菜单项的右边，下拉菜单会插入到当前选择菜单的下面。选择 Delete 会删除一个菜单。
- 如果要插入一个子菜单，则可以从弹出的快捷菜单中选择 Create SubMenu。
- 要添加菜单执行的命令，可以通过为该菜单设置 OnClick 事件实现。
- 要改变图形的形状，可通过其 Shape 属性设置不同的值实现。修改 Brush 和 Pen 属性的值，可以改变内部及轮廓的颜色及风格。
- “颜色”对话框使用 Color 属性表示当前选择的颜色。另外，可通过将颜色值右移位并进行“或”运算，获得 R、G、B 三种颜色的混合结果。
- 要调用另一个单元文件定义的变量或函数等，应使用 uses 语句。
- 可以通过使用 Sender 对象获得界面上当前选择的对象。



### 归纳注释

设置菜单主要使用如下步骤：首先通过 MainMenu 或 PopuMenu 组件添加使用的菜单项，要在菜单编辑器中右击并通过 Insert 菜单插入新的菜单项；然后为各个菜单设置 OnClick 事件。

如果是通过 `ActionList` 组件设置各个菜单实现的功能，就不需要为各个菜单项单独设置 `OnClick` 事件，而应通过 `Action` 属性设置 `ActionList` 组件中定义的合适动作。

最后，要使用定义的菜单，应设置表单或其他组件的 `Menu` 及 `PopupMenu` 属性，并将其设置为新定义的主菜单或弹出菜单的名字。

该实例包括两个表单 `MenuForm` 和 `AboutBox`，所以对应两个单元文件。这两个单元文件的源代码如下：

(1) 表单 `MenuForm` 对应的单元文件 `menudemo`:

```
unit menudemo;
...           //略

const
  SHAPE_OFFSET = 4;
var
  MenuForm: TMenuForm;

implementation

uses about;           //调用表单 AboutBox 的单元文件 about

{$R *.DFM}

procedure TMenuForm.DisplayPanelResize(Sender: TObject); //改变显示面板大小
begin
  DemoShape.Top := SHAPE_OFFSET;
  DemoShape.Left := DemoShape.Top;
  DemoShape.Height := DisplayPanel.Height - 2 * SHAPE_OFFSET;
  DemoShape.Width := DisplayPanel.Width - 2 * SHAPE_OFFSET;
end;

procedure TMenuForm.Exit1Click(Sender: TObject); //退出菜单
begin
  Close;
end;

function TMenuForm.RandomColor; //生成一种任意颜色
var
  red, green, blue: Byte;
begin
  red := Random(255);
  green := Random(255);
  blue := Random(255);

  Result := red or (green shl 8) or (blue shl 16);
end;
```



```
procedure TMenuForm.Randomize1Click(Sender: TObject); //单击“轮廓|任意”菜单
begin
    DemoShape.Pen.Color := RandomColor;
end;

procedure TMenuForm.Select1Click(Sender: TObject); //单击“轮廓|选择”菜单
begin
    try
        SolidColorDialog.Color := DemoShape.Pen.Color;
        if SolidColorDialog.Execute then
            DemoShape.Pen.Color := SolidColorDialog.Color;
    except
        ShowMessage('无法打开选择的颜色对话框!');
    end;
end;

procedure TMenuForm.Randomize2Click(Sender: TObject); //单击“内部|任意”菜单
begin
    DemoShape.Brush.Color := RandomColor;
end;

procedure TMenuForm.Select2Click(Sender: TObject); //单击“内部|选择”菜单
begin
    AnyColorDialog.Color := DemoShape.Brush.Color;
    try
        if AnyColorDialog.Execute then
            DemoShape.Brush.Color := AnyColorDialog.Color;
    except
        ShowMessage('无法打开选择的颜色对话框!');
    end;
end;

procedure ToggleCheck(Sender: TObject); //设置“切换”复选框
var
    Item: TMenuItem;
begin
    Item := Sender as TMenuItem;
    Item.Checked := not Item.Checked;
end;

procedure SetCheck(Sender: TObject); //设置复选框
var
    Item: TMenuItem;
begin
    Item := Sender as TMenuItem;
    Item.Checked := True;
end;
```