

编程实例教程系列

Delphi 7 编程 实例教程

刘刚 陈蓓 等编

兵器工业出版社



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

编程实例教程系列

Delphi 7 编程 实例教程

刘刚 陈蓓 等编

兵器工业出版社



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

内 容 简 介

本书介绍使用 Delphi 进行实用程序开发过程中所遇到的各种问题。在介绍编程原理的同时, 利用大量的实例总结编程经验, 内容包括: 界面设计、文件应用、多媒体设计、网络应用、打印程序的开发、物理信息的获取、数据库应用等。在每一部分中都有程序清单和注释。

本书以一种清晰而简练的风格介绍 Delphi 的使用和技巧, 通过实例演示正确的使用方法和设计习惯。本书适合于具有基本的面向对象思想基础的读者, 同时也可作为高校相关专业教学和自学参考书。

图书在版编目 (CIP) 数据

Delphi 7 编程实例教程/刘刚, 陈蓓等编. —北京: 兵器工业出版社; 北京希望电子出版社, 2004. 12
(编程实例教程系列)
ISBN 7-80172-279-5

I. D... II. ①刘... ②陈... III. 软件工具—程序设计—教材 IV. TP311. 56

中国版本图书馆 CIP 数据核字 (2004) 第 074274 号

出 版: 兵器工业出版社 北京希望电子出版社
邮编社址: 100089 北京市海淀区车道沟 10 号
100085 北京市海淀区上地信息产业基地 3 街 9 号
金隅嘉华大厦 C 座 610
发 行: 北京希望电子出版社
电 话: (010) 82702660 (发行) (010) 62541992 (门市)
经 销: 各地新华书店 软件连锁店
印 刷: 北京媛明印刷厂
版 次: 2004 年 12 月第 1 版第 1 次印刷

封面设计: 梁运丽
责任编辑: 宋丽华 安 源
责任校对: 周凤明
开 本: 787×1092 1/16
印 张: 24
印 数: 1-5000
字 数: 557 千字
定 价: 35.00 元

(版权所有 翻印必究 印装有误 负责调换)

前 言

随着计算机技术的飞速发展，计算机的开发迅速得到普及和推广。当前有一大批计算机开发人员，虽然已经掌握了某种开发工具的基础知识，甚至达到了相当高的水平，但仍然有很多盲区，急需更多的编程经验和技巧。

笔者喜爱 Delphi 的原因很简单，就是因为面板上的组件。VC 的 MFC 难以做出美观的界面，VB 的语法太松散，笔者都不是很喜欢。随着实践经验的不断增加，越来越感到 Delphi 的魅力，其博大精深是 VB 无法相比的，而且毫不逊色于 VC。作为流行的 RAD 工具之一，Delphi 真正体现了使用简便和开发快速的特点。如果用户需要做非常底层的应用，那么 Delphi 也同样是一个好选择。它几乎对 Windows 所有的 API 都进行了封装，OPENGL 单元还提供了大量的涉及 OpenGL 编程的函数，用户可以直接在程序中嵌入汇编代码段，一方面可以使程序直接操作硬件；另外一方面，即使用户不操作硬件，汇编的使用也使程序代码的执行效率大为提高。如果既要美观的界面又对底层进行操作，那么 Delphi 是最好的选择。从软件公司的需求来看，越来越多的软件公司都选择了 Delphi 作为 Windows 主要的编程工具。“真正的程序员用 C，聪明的程序员用 Delphi”，这句话是很有道理的。

书中的实例取材于笔者的开发实例和网上资源。本书内容涉及实际开发过程中遇到的各个方面，相信读者一定可以从本书中学到所需的知识。

本书的内容涉及：界面设计，文件应用，多媒体设计，网络应用，打印程序的开发，物理信息的获取，数据库应用，高级编程。在每一部分中都有程序的清单，在程序中对比较重要的代码段都做了注释，以使读者一目了然，冗长之处，还请读者谅解。

为了帮助读者更好地领会这些技巧，我们建议读者：先看一遍代码，抓住每个实例中的关键技巧，然后运行完整的源程序，看看程序具体的运行结果；如果已经掌握了，那么请设计类似的程序来检验自己的理解程度；如果能对实例程序进行修改那就再好不过了。

本书由刘刚和陈蓓组织编写，其他参加本书编写、录排、校对工作的人员还有：田丽楹、田军、郭祥雷、赵文辉、黄晓润、田野、赵军锁、姜南、车自友、张文敏、强秀丽、龚志翔、龚建、白红利、葛丽、汪建华、刘聪、邓波、王东霞等。龚超同志对全稿进行了严格细致的复审。本书的出版是集体劳动的结晶，在此特别感谢北京希望电子出版社第三编辑室的编辑们，他们为本书的出版付出了很多心血！

由于时间和作者的水平有限，错误在所难免，如果有读者有疑问之处，欢迎进行咨询；如果读者对书中的例子有更好的实现方法，也欢迎与笔者探讨，以让我们共同进步。联系方式：admin_mail@163.com。最后，笔者把本书送给我的好朋友 YaYe，并祝愿 YaYe 早日康复。

笔 者

目 录

第1章 界面设计.....	1	2.2.4 说明.....	62
1.1 概述.....	1	2.3 在 Delphi 中调用 Word 文件.....	62
1.2 启动界面程序.....	1	2.3.1 相关组件和函数介绍.....	62
1.2.1 相关组件和函数介绍.....	2	2.3.2 开发步骤.....	63
1.2.2 开发步骤.....	3	2.3.3 程序清单.....	74
1.2.3 程序清单.....	5	2.3.4 开发步骤.....	75
1.2.4 说明.....	8	2.3.5 程序清单.....	76
1.3 基于 GUI 的悬浮式菜单程序.....	8	2.3.6 说明.....	78
1.3.1 相关组件和函数介绍.....	8	2.4 读写 INI 文件.....	78
1.3.2 开发步骤.....	12	2.4.1 相关组件和函数介绍.....	79
1.3.3 程序清单.....	14	2.4.2 开发步骤.....	84
1.3.4 说明.....	17	2.4.3 程序清单.....	85
1.4 将应用程序的图标添加到 任务栏中状态区程序.....	17	2.4.4 说明.....	87
1.4.1 相关组件和函数介绍.....	17	2.5 将数据写入到 HTML 中.....	87
1.4.2 开发步骤.....	20	2.5.1 相关组件和函数介绍.....	87
1.4.3 程序清单.....	21	2.5.2 开发步骤.....	88
1.4.4 说明.....	24	2.5.3 程序清单.....	90
1.5 屏幕保护预览程序.....	24	2.5.4 说明.....	92
1.5.1 相关组件和函数介绍.....	25	2.6 操作注册表文件.....	92
1.5.2 开发步骤.....	26	2.6.1 相关组件和函数介绍.....	93
1.5.3 程序清单.....	30	2.6.2 开发步骤.....	97
1.5.4 说明.....	36	2.6.3 程序清单.....	99
1.6 安装程序.....	36	2.6.4 说明.....	101
1.6.1 相关组件和函数介绍.....	37	2.7 本章小结.....	101
1.6.2 开发步骤.....	39	第3章 多媒体设计.....	102
1.6.3 程序清单.....	46	3.1 概述.....	102
1.6.4 说明.....	52	3.2 Delphi 实现多媒体视频播放.....	102
1.7 本章小结.....	52	3.2.1 相关组件和函数介绍.....	102
第2章 文件应用.....	53	3.2.2 开发步骤.....	104
2.1 概述.....	53	3.2.3 程序清单.....	107
2.2 将数据导入 EXCEL 文件.....	53	3.2.4 说明.....	110
2.2.1 相关组件和函数介绍.....	53	3.3 判断系统中安装的声卡.....	110
2.2.2 开发步骤.....	55	3.3.1 相关组件和函数介绍.....	110
2.2.3 程序清单.....	58	3.3.2 开发步骤.....	111
		3.3.3 程序清单.....	111

3.3.4 说明.....	113
3.4 Delphi 实现多媒体音频播放	113
3.4.1 相关组件和函数介绍.....	113
3.4.2 开发步骤.....	115
3.4.3 程序清单.....	118
3.4.4 说明.....	122
3.5 弹出、关闭 CD-ROM	122
3.5.1 相关组件和函数介绍.....	122
3.5.2 开发步骤.....	123
3.5.3 程序清单.....	124
3.5.4 说明.....	125
3.6 实现屏幕抓取	125
3.6.1 相关组件和函数介绍.....	126
3.6.2 开发步骤.....	129
3.6.3 程序清单.....	135
3.6.4 说明.....	142
3.7 本章小结	142
第 4 章 打印程序的开发与设计	143
4.1 概述	143
4.2 使用 Delphi 获取打印机设备 名和端口名	143
4.2.1 相关组件和函数介绍.....	143
4.2.2 开发步骤.....	145
4.2.3 程序清单.....	147
4.2.4 说明.....	149
4.3 编写打印驱动程序	149
4.3.1 相关组件和函数介绍.....	149
4.3.2 开发步骤.....	151
4.3.3 程序清单.....	170
4.3.4 说明.....	172
4.4 用 API 直接打印	172
4.4.1 相关组件和函数介绍.....	172
4.4.2 开发步骤.....	174
4.4.3 程序清单.....	175
4.4.4 说明.....	177
4.5 本章小结	177
第 5 章 网络应用	178
5.1 概述	178
5.2 通过 IP 地址获得计算机名称.....	178

5.2.1 相关组件和函数介绍.....	178
5.2.2 开发步骤.....	179
5.2.3 程序清单.....	180
5.2.4 说明.....	182
5.3 获得 MAC 地址和子网掩码, 以及 默认网关和网卡品牌	182
5.3.1 基本概念.....	182
5.3.2 相关组件和函数介绍.....	184
5.3.3 开发步骤.....	184
5.3.4 程序清单.....	186
5.3.5 说明.....	187
5.4 编程实现 Ping 操作	188
5.4.1 Ping 命令简介	188
5.4.2 相关组件和函数介绍.....	189
5.4.3 开发步骤.....	190
5.4.4 程序清单.....	192
5.4.5 说明.....	196
5.5 检测计算机与 Internet 连接状态	196
5.5.1 相关组件和函数介绍.....	197
5.5.2 开发步骤.....	197
5.5.3 程序清单.....	199
5.5.4 说明.....	203
5.6 获得网络邻居中的所有的工作组	203
5.6.1 相关组件和函数介绍.....	204
5.6.2 开发步骤.....	208
5.6.3 程序清单.....	211
5.6.4 说明.....	213
5.7 测试网络网速	214
5.7.1 相关组件和函数介绍.....	214
5.7.2 开发步骤.....	215
5.7.3 程序清单.....	216
5.7.4 说明.....	218
5.8 获取与远程机连接时所经过的 路由信息	218
5.8.1 相关组件和函数介绍.....	218
5.8.2 开发步骤.....	219
5.8.3 程序清单.....	220
5.8.4 说明.....	222
5.9 构建 FTP 程序	223

5.9.1 什么是 FTP.....	223	6.6.1 相关组件和函数介绍.....	264
5.9.2 相关组件和函数介绍.....	223	6.6.2 开发步骤.....	265
5.9.3 开发步骤.....	225	6.6.3 程序清单.....	267
5.9.4 程序清单.....	226	6.6.4 说明.....	270
5.9.5 说明.....	228	6.7 程序注册码设计.....	270
5.10 实现 EMAIL 的发送.....	228	6.7.1 相关组件和函数介绍.....	270
5.10.1 相关组件和函数介绍.....	230	6.7.2 开发步骤.....	274
5.10.2 开发步骤.....	232	6.7.3 程序清单.....	277
5.10.3 程序清单.....	233	6.7.4 说明.....	282
5.10.4 说明.....	238	6.8 本章小结.....	282
5.11 信使服务的实现.....	238	第 7 章 数据库编程	283
5.11.1 相关组件和函数介绍.....	238	7.1 概述.....	283
5.11.2 开发步骤.....	239	7.2 将图片存入数据库.....	283
5.11.3 程序清单.....	242	7.2.1 相关组件和函数介绍.....	283
5.11.4 说明.....	245	7.2.2 开发步骤.....	292
5.12 本章小结.....	245	7.2.3 程序清单.....	294
第 6 章 物理信息的获取	246	7.2.4 说明.....	298
6.1 概述.....	246	7.3 自动在 ODBC 中配置数据库源.....	298
6.2 测试 CPU 的速度.....	246	7.3.1 相关组件和函数介绍.....	299
6.2.1 相关组件和函数介绍.....	246	7.3.2 开发步骤.....	300
6.2.2 开发步骤.....	247	7.3.3 程序清单.....	303
6.2.3 程序清单.....	248	7.3.4 说明.....	306
6.2.4 说明.....	251	7.4 创建一个数据模块.....	307
6.3 获取物理内存.....	251	7.4.1 相关组件和函数介绍.....	307
6.3.1 相关组件和函数介绍.....	251	7.4.2 开发步骤.....	313
6.3.2 开发步骤.....	252	7.4.3 程序清单.....	315
6.3.3 程序清单.....	253	7.4.4 说明.....	317
6.3.4 说明.....	255	7.5 编写一个查询程序.....	318
6.4 获得当前显示分辨率.....	255	7.5.1 相关组件和函数介绍.....	318
6.4.1 相关组件和函数介绍.....	255	7.5.2 开发步骤.....	319
6.4.2 开发步骤.....	257	7.5.3 程序清单.....	320
6.4.3 程序清单.....	258	7.5.4 说明.....	321
6.4.4 说明.....	259	7.6 配置小型的数据库引擎.....	321
6.5 获取系统颜色数目.....	260	7.6.1 相关组件和函数介绍.....	322
6.5.1 相关组件和函数介绍.....	261	7.6.2 开发步骤.....	323
6.5.2 开发步骤.....	262	7.6.3 说明.....	323
6.5.3 程序清单.....	262	7.7 本章小结.....	323
6.5.4 说明.....	263	第 8 章 高级编程技术	324
6.6 找出磁盘上任何子目录下的文件.....	264	8.1 概述.....	324

8.2 多线程的同步	324
8.2.1 相关组件和函数介绍	326
8.2.2 开发步骤	328
8.2.3 程序清单	330
8.2.4 说明	334
8.3 DLL 的应用	334
8.3.1 相关组件和函数介绍	337
8.3.2 开发步骤	338
8.3.3 程序清单	340
8.3.4 说明	342
8.4 数据写入 XML 文件	343
8.4.1 相关组件和函数介绍	344
8.4.2 开发步骤	348

8.4.3 程序清单	351
8.4.4 说明	354
8.5 实现 COM	355
8.5.1 相关组件和函数介绍	355
8.5.2 开发步骤	356
8.5.3 程序清单	357
8.5.4 说明	364
8.6 实现 CORBA	364
8.6.1 相关组件和函数介绍	366
8.6.2 开发步骤	369
8.6.3 程序清单	370
8.6.4 说明	375
8.7 本章小结	375

第 1 章 界面设计

主要内容:

- 概述
- 启动界面程序
- 基于 GUI 的悬浮式菜单程序
- 将应用程序的图标添加到任务栏中状态区程序
- 屏幕保护预览程序
- 安装程序
- 本章小结

1.1 概 述

计算机用户界面是指计算机与使用者之间的对话接口,是计算机系统重要的组成部分。计算机的发展不仅是处理速度和存储容量飞速提高的历史,而且还包括用户界面不断改进的过程。早期计算机是通过面板上的指示灯来显示二进制数据和指令的,那时人们通过开关、按键以及穿孔纸带输入和显示各种数据及命令。到 20 世纪 50 年代中、后期,由于采用了作业控制语言(JCL)和控制台打字机,使得计算机可以批处理多个计算任务,从而代替原来笨拙的手工按键方式,提高了计算机的使用效率。1963 年,美国麻省理工学院在 709/7090 计算机的基础上成功地开发出第一个分时系统 CTSS,它连接了多个分时终端,并最早使用了文本编辑程序。从此,以命令行形式对话的多用户分时终端成为 20 世纪 70~80 年代计算机用户界面的主流方式。20 世纪 80 年代初,美国 Xerox 公司的 Alto 计算机首先使用了 Smalltalk-80 程序设计开发环境,以及后来的 Lisa, Macintosh 等计算机,将用户界面推向图形用户界面的新阶段,随之而来的用户界面管理系统和智能界面的研究均推动了用户界面的发展。用户界面已经从过去的人去适应笨拙的计算机,发展到今天的计算机不断地适应人的需求。

用户界面的重要性在于它极大地影响了最终用户的使用、计算机的推广应用,甚至是人们的工作和生活。由于开发用户界面的工作量极大,加上不同用户对界面的要求也不尽相同,所以用户界面已成为计算机软件研制中最困难的部分之一。当前 Internet 的发展异常迅猛,加上虚拟现实,科学计算可视化以及多媒体等技术都对用户界面提出了更高的要求。

本章所提供的 5 个实例都是在应用开发过程中常见功能的实现,为了便于读者理解相关的技术,每个例子中将介绍相关组件和函数的基本概念。

1.2 启动界面程序

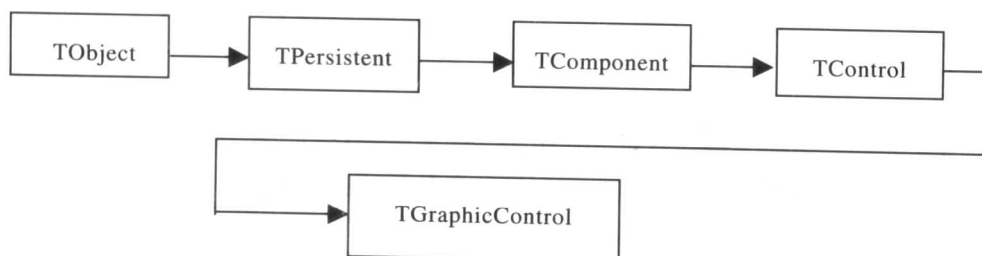
通常 Windows 程序显示一个启动提示或者界面,它也叫做闪现窗口。在创建时

由于工程本身创建了程序的主窗口，所以一般情况下主窗口是最先出现的，那么闪现窗口如何出现在主窗口之前呢？假设将主窗口的可视属性（Visible）设置为假，当工程启动时，是否能将主窗口暂时隐藏起来，以使闪现屏幕最先出现？答案是不能！因为应用程序自动在适当时候使 Form 可视(当使用 Form 作为子窗口或对话框，如果不需要，那么将可视属性设为假，则可隐藏 Form)。那么如何解决这一问题呢，通过下面的实例我们来实现启动界面。

1.2.1 相关组件和函数介绍

TImage 组件介绍

TImage 组件在 Additional 组件页中，是图像显示组件。用于显示图、图标或媒体文等，其层次关系如下所示。



在窗体中使用 Timage 组件可以显示一幅图像，要显示的图像使用 Tpicture 对象指定。Timage 引入的一些属性确定 Timage 对象边界内如何显示图像。要在窗体或数据模块上添加一幅其他控件用的图像，如行为列表和位图按钮，可以使用 TimageList 组件。

(1) 属性

Timage.Picture

声明: `property Picture: Tpicture;`

属性指定在 Image 组件中显示的图像，即 Tpicture 对象，其中包含图标或位图图像。属性的图像可以在图像编辑器中调入。

Timage.Transparent

声明: `property Transparent: Boolean;`

属性指定图像的背景是否遮盖其下面的图像对象。为 True，可以透过位图的背景看到 Timage 对象下面的对象。为 False，位图背景不透明。设置属性就设置了 Picture 的 Transparent 属性。

Timage.Stretch

声明: `property Stretch: Boolean;`

属性指定 Image 组件中的图像是否可以改变，以确切地将图像放入图像的边界。为 True，图像适合图像控件的大小和形状。当图像控件大小改变时，图像大小也改变。该属性改变图像的高和宽，因此，在图像控件和图像形状不一样时，它可以扭曲图像控件上的图像。要调整 Image 组件中图像与组件的大小相同，最好使用

Autosize 属性。如果 Picture 属性包含的是一个图标，则该属性无效。

(2) 方法

Timage.Timage()

声明: constructor Create(Aowner:Tcomponent);override;

方法创建并初始化一个 Timage 的实例。程序运行时，调用该方法可以实例化一个 Timage 对象；程序设计时，在窗体中放置 Timage 组件后自动创建该组件的实例。

Timage.Destroy()

声明: destructor Destroy;override;

方法销毁 Timage 实例。应用程序中不能直接调用该方法，而应该先调用 Free，以检查调用 Destroy 前 Timage 参数不为 0，然后调用该方法。

(3) 事件

Timage.OnProgress

声明: type TprogressStage=(psStarting,psRunning,psEnding);
TprogressEvent=procedure(Sender:Tobject;Stage:TprogressStage;PercentDone:Byte;
RedrawNow:Boolean;const R:Trect;const Msg:String) of object;
Property OnProgress:TprogressEvent

图像显示效果在慢速操作期间定时触发 OnProgress 事件，编写这个事件的处理程序可以响应图像显示的慢速操作。参数 Stage 表示操作开始或结束。如果这个事件的处理程序显示一个指示器时，指示器取决于 Stage 参数的值。Stage 为 psStarting 时产生指示器，为 psRunning 时更新指示器，为 psEnding 时删除指示器。PercentDone 参数指定已经完成操作的比例，用其可以更新进程状态栏或其他指示器的位置。RedrawNow 参数指定图像是否在屏幕上安全地绘图。

1.2.2 开发步骤

(1) 开始一个新工程 Form1，并且新添加一个窗体 Form2，在如图 1-1 所示的图形窗体定制界面，将窗体的 BorderStyle 属性设置 bsNone，并在其中添加一个 Image 组件。

(2) 在 Form1 的 OnActivate 事件和 OnCreate 事件中添加如下代码。

```
Procedure TForm1.FormActivate(Sender: TObject);
Begin
  Form2.Showmodal();
End;
```

(3) 在 Form 的 OnCreate 事件中添加如下代码。

```
Procedure TForm1.FormCreate(Sender: TObject);
var
```

```

CurTime:LongWord ;
DelayTime:LongWord;
Begin
// 设置延迟时间为 5 秒
DelayTime:=4000;
CurTime:=GetTickCount ;
while (GetTickCount<(CurTime+DelayTime)) do ;
// 在这儿可以加上应用程序的初始化过程
End;

```

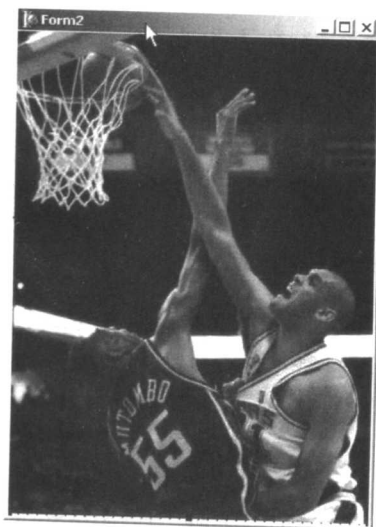


图 1-1 设计界面

(4) 在工程文件中更改如下代码。

```

program Project1;
uses Forms,
Unit1 in 'Unit1.pas' {Form1},
Unit2 in 'Unit2.pas' {Form2};
{$R *.res}
Begin
Try
Form2:=TForm2.Create(Application); // 创建 Form2
Form2.Show ; // 显示 Form2
Form2.Update ; // 更新 Form2
Application.Initialize;
Application.CreateForm(TForm1, Form1);

```

```
finally
Form2.Free ;
    End;
End.
```

(5) 在 From2 的 FormClick 事件中添加如下代码。

```
procedure TForm2.FormClick(Sender: TObject);
begin
Close;
end;
```

(6) 程序启动运行后的结果如图 1-2 所示。

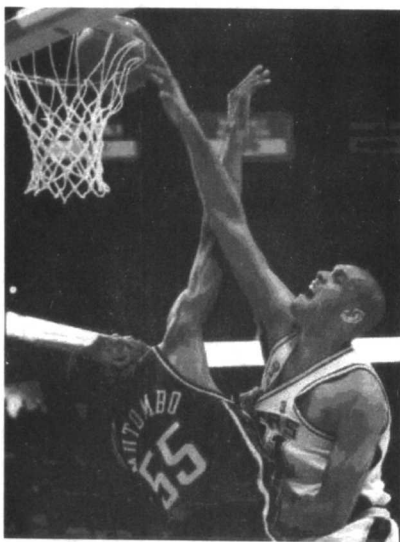


图 1-2 启动后的界面

1.2.3 程序清单

下面给出各部分的清单。

unit1 的程序清单

```
unit Unit1;
interface
uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
    Dialogs;
type
    TForm1 = class(TForm)
```

```
    procedure FormActivate(Sender: TObject);
    procedure FormCreate(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

var
    Form1: TForm1;

implementation
uses Unit2;
{$R *.dfm}
procedure TForm1.FormActivate(Sender: TObject);
begin
    form2.showmodal();
end;

procedure TForm1.FormCreate(Sender: TObject);
var
    CurTime: LongWord ;
    DelayTime: LongWord;
begin
    // 设置延迟时间为 5 秒
    DelayTime:=4000;
    CurTime:=GetTickCount ;
    while (GetTickCount<(CurTime+DelayTime)) do ;
    // 在这儿可以加上应用程序的初始化过程
end;

end.
```

unit2 的程序清单

```
unit Unit2;

interface
```



```

uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
    Dialogs, ExtCtrls, jpeg;
type
    TForm2 = class(TForm)
        Image1: TImage;
        procedure FormClick(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;
var
    Form2: TForm2;

implementation

{$R *.dfm}
procedure TForm2.FormClick(Sender: TObject);
begin
    close;
end;
end.

```

主程序清单

```

program Project1;
uses
    Forms,
    Unit1 in 'Unit1.pas' {Form1},
    Unit2 in 'Unit2.pas' {Form2};

{$R *.res}

```

```
begin
  try
    Form2:=TForm2.Create(Application); // 创建 Form2
    Form2.Show ; // 显示 Form2
    Form2.Update ; // 更新 Form2
    Application.Initialize;
    Application.CreateForm(TForm1, Form1);
  finally
    Form2.Free ; //
  end;
end.
```

1.2.4 说明

实例可以做到屏幕上显示出一个闪现窗口，从中可以了解 From 的动态创建和撤销的用法，根据具体的情况灵活使用一定能给程序添色不少。实例对应的源程序可以从光盘中的 Chapter1\1 中找到。

1.3 基于 GUI 的悬浮式菜单程序

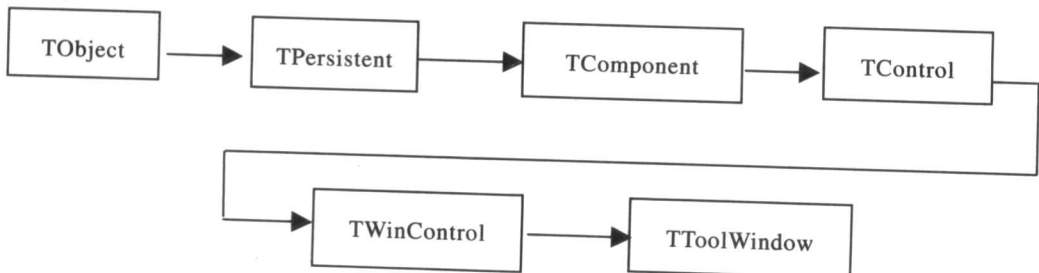
图形用户界面（GUI）的广泛流行是当今计算机技术的重大成就之一，它极大方便了非专业用户的使用。人们不再需要死记硬背大量的命令，而可以通过窗口、菜单方便地进行操作。用过 Word 的朋友都知道，它的 TOOLBAR 是可以在屏幕上自由拖动的，那么如何实现呢？下面来编写一个有关的实例，可以拖放的菜单工具栏和普通工具栏，可在 Form 的四周停靠，并可在程序启动时浮动在桌面（不停靠）。

1.3.1 相关组件和函数介绍

TToolBar 组件和 TToolBar 组件位于 Win32 组件页。

1. TToolBar 组件介绍

TToolBar 组件用于管理工具和其他组件，其层次关系如下所示。



(1) 属性

TToolBar.ButtonCount

声明: `property ButtonCount:integer;`

属性指定工具栏中按钮的数量,即 ButtonCount 在 Buttons 数组中给出了按钮的数量。

TToolBar.Canvas

声明: `property Canvas:Tcanvas;`

在 OnCustomDraw 或 OnCustomDrawItem 事件处理程序中,使用属性可在工具栏表面绘制。TCanvas 对象为绘制图像提供了属性和方法。

TToolBar.Flat

声明: `property Flat:Boolean;`

属性为 True,工具栏及工具按钮是透明的,并且单击按钮后,工具栏上的工具按钮显示弹出的边框。此时在程序设计和运行时,TToolBar 需要 COMCTL32.DLL 4.70 或更高的版本。

TToolBar.Wrapable

声明: `property Wrapable:Boolean;`

属性允许工具栏上的组件自动换行。为 True,当工具栏上的组件无法在水平方向放置时自动换行。此外,从 TToolWindow 派生的属性如下。

TToolWindow.EdgeBorders:TedgeBorders

声明: `property EdgeBorders:TedgeBorders;`

使用该属性可以添加或删除边框,边框的风格用 EdgeInner 和 EdgeOuter 属性设置。

TToolWindow.EdgeInner

声明: `property EdgeInner:TedgeStyle;`

属性确定组件内部边缘的边框风格,取值见表 1-1 所示。

表 1-1 内部边缘的边框风格

值	说明
EsNone	内部边缘无边框
EsRaised	内部边缘凹起
EsLowered	内部边缘凸下

EdgeInner 和 EdgeOuter 属性可以使组件产生 3D 效果。

TToolWindow.EdgeOuter:TedgeStyle;

声明: `property EdgeOuter:TedgeStyle;`

属性确定组件的外部边缘的边框风格,取值如表 1-2 所示。