

商业开发 代码库系列

C++ Builder

案例开发集锦

刘超 唐彬 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

商业开发代码库系列

C++ Builder案例开发集锦

刘 超 唐 彬 编著

電子工業出版社.

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书利用有代表性、实用性强、效果新颖的具体案例讲解了C++ Builder 6.0在控件、数据库、多媒体、游戏、图形图像、文件处理、网络及API函数等方面的具体应用,基本涵盖了C++ Builder所有的应用领域。本书案例经典、内容全面、技术实用、资源丰富,每一个案例都是精心挑选的,实用性强,解释详尽,随书附有光碟源代码,让读者在实例中学习编程思想和编程技巧。

本书不仅适用于想进入编程领域的读者和计算机专业的大、中专学生阅读,对于计算机专业人士也有较高的参考价值。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

C++ Builder案例开发集锦/刘超等编著. —北京:电子工业出版社, 2005.8

(商业开发代码库系列)

ISBN 7-121-01462-9

I. C… II. 刘… III. C语言—程序设计 IV. TP312

中国版本图书馆CIP数据核字(2005)第068064号

责任编辑:朱 巍

印 刷:北京天竺颖华印刷厂

出版发行:电子工业出版社

北京市海淀区万寿路173信箱 邮编:100036

北京市海淀区翠微东里甲2号 邮编:100036

经 销:各地新华书店

开 本:787×1092 1/16 印张:25.5 字数:650千字

印 次:2005年8月第1次印刷

定 价:40.00元(含光碟1张)

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换,若书店售缺,请与本社发行部联系。联系电话:010-68279077。质量投诉请发邮件至zlts@phei.com.cn,盗版侵权举报请发邮件至dbqq@phei.com.cn。

前 言



我们学习了一门计算机语言之后，总想利用它编写出一些精彩的程序。许多读者希望快速获得一些关于编程的宝贵经验，迅速达到“高手”的境界。然而，精彩的程序背后是无数经验的积累。

C++ Builder是Borland (Inprise) 公司于1998年推出的全新32位Windows开发工具，是至今最好的Windows开发工具之一。它集编程、调试、运行于一体，初学者一看就懂，一用就明白，节省了入门时间。C++ Builder 6.0提供面向对象及可视化设计的快速应用程序开发环境，使用C++ Builder 6.0可以用较少程序代码编写出各类复杂且高性能的Windows应用程序。C++ Builder 6.0为开发者提供功能强大的VCL，方便结合和扩展已有控件，提供高效能数据库访问，支持网络编程，编译速度和执行速度更快。

本书针对使用C++ Builder 6.0开发软件的各个层次的读者。全书分9章，每章从多个方面，通过实用性较高和技术性较强的实例，介绍C++ Builder 6.0编程方法与技巧。

第1章 C++ Builder与控件编程

本章主要介绍了构造基本Windows应用程序的一些常规组件的使用技巧，包括介绍组件的一些常用属性和方法。通过对本章的学习，读者能够掌握常用Windows组件的使用方法，学会处理一些技巧方面的编程。另外，本章介绍的组件只是一小部分，C++ Builder还提供了许多其他组件，在本书以后的章节中我们也会陆续接触到。

第2章 C++ Builder数据库编程

本章主要介绍了C++ Builder在数据库方面的程序设计，详细介绍了C++ Builder提供的各种数据库组件，涉及到多种数据库的连接和操作，包括Access、SQL Server 2000等；而且在数据库操作方面介绍了多种数据库的连接方式、操作方式；介绍了我们经常使用的数据集和数据感知组件；还通过简单报表的设计介绍了使用QuickReport模板创建报表的方法。本章通过丰富的实例代码说明了各个数据库组件的使用方法，通过对本章的学习，读者可以自行设计一些相对复杂的数据库应用程序。

第3章 C++ Builder的图像处理

本章主要介绍了对图像的基本操作，包括移动、旋转、色彩变化以及实现图像动画的一些方法；另外还介绍了实现以上效果用到的一些组件、API函数等；此外还介绍了组件的属性、方法。通过对本章的学习，读者对C++ Builder的图像组件有一定的了解。本章实例代码丰富，简单易懂，能够使读者感到图像的变幻，动画的实现是如此地简单。

第4章 C++ Builder多媒体与游戏编程

本章主要介绍了一些C++ Builder的编程技巧、多媒体组件的应用以及ActiveX的装载，在介绍MediaPlayer组件的基础上，详细地讨论设计多媒体应用程序的方法，并对MCI做了简

单的分析。通过对本章的学习，读者可以感受到多媒体播放软件的设计是如此简单，因为C++ Builder提供了功能强大的多媒体组件，读者可以用简单的代码来实现强大的功能。

第5章 C++ Builder网络编程

本章详细说明了C++ Builder多种Internet组件的属性、事件和方法，包括CppWeb-Browser、NMHTTP、SocketClient、SocketServer等组件，进一步讲解了它们的使用方法，并通过详细的代码实例再次说明了各种组件的实际应用。通过对本章的学习，读者对Internet应用程序的设计应有大体的了解，可以使读者感觉到Internet应用程序不是那么神秘，我们也可以设计出自己的Internet应用程序。

第6章 C++ Builder文件处理

本章主要介绍了VCL提供的有关目录和文件处理的函数和类，通过丰富的实例、简单的代码说明如何使用这些函数和类。通过对本章的学习，读者应该可以掌握文件处理的一些基本方法，并能设计自己的文件编辑应用程序。

第7章 C++ Builder消息API编程

本章主要介绍了多种API函数的原型及其功能，讲解了如何将API应用到多种应用程序中去。API设计的范围较广，在各种应用程序中我们都可以应用API函数，但是API函数相对较为复杂，通过对本章的学习，读者可以对常用的API函数有一定的了解，而且应该能够掌握API函数在应用程序中的应用。

第8章 其他应用实例

前面的章节已经涉及到了C++ Builder的多个方面，本章主要通过一些简单实例来了解C++ Builder其他方面的编程，包括DLL的调用、COM技术的应用和托盘编程等，这些在应用程序的开发中应用也比较广泛，许多读者可能不太了解。通过对本章的学习，读者可以对DLL、COM等有一个初步的了解，学会基于COM技术的应用程序设计。

第9章 综合实例

通过前面的学习，读者已经有了一定的基础，本章主要通过几个实例来对前面所讲的C++ Builder知识做个总结，这几个实例涉及的面比较广，包括了控件、文件、数据、网络等。通过对本章的学习，读者应该对实际应用程序的开发有一定的了解。

最后，感谢和我共同完成此书的合作者，他们是：冯韬略、王艳伟、徐霞、霍东升、缪素芬、李强丽、高立刚、齐凯。感谢北京美迪亚电子信息有限公司的各位老师，谢谢你们的帮助和指导。由于我们水平有限，对书中不妥之处，恳请读者批评指正。

目 录

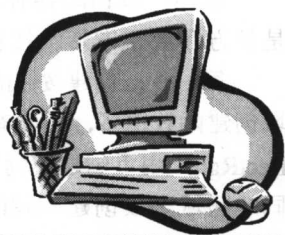


第1章 C++ Builder与控件编程	1
圆形、三角形按钮的实现	1
用C++ Builder创建组合框	3
C++ Builder动态建立菜单	5
身份证号识别程序	8
C++ Builder制作闹钟	10
中英文输入法的自动切换	13
ListBox自绘	16
在RichEdit中实现上标下标	21
制作主窗口显示前的版权窗口	23
C++ Builder中不规则窗体的快速显示	25
动态生产按钮对象	30
C++ Builder中保持控件的位置和大小	33
Memo技巧编程	36
ListView点击排序	38
 第2章 C++ Builder数据库编程	 41
连接数据库	41
OLE控件编程Excel	45
Access版本密码	51
动态创建数据源	56
打印预览	59
表的创建和添加	67
ADOTable的表操作	72
通讯录	82
SQL命令数据库操作	88
用户信息查询	91
 第3章 C++ Builder的图像处理	 96
大尺寸图像的显示技巧	96
特殊的屏幕显示效果	99

个性凸现的动画特技效果	109
显示透明位图	116
位图旋转	119
实现窗体的渐变背景	121
实现快速动画	125
从系统中取得指定资源图像	127
颜色渐变的进度条	130
第4章 C++ Builder多媒体与游戏编程	133
悬浮窗口的实现	133
Winnap磁性窗口的建立	135
播放Flash	137
多任务多媒体的播放	141
多媒体计时器	144
实现动画	146
发送图像文件	152
九宫游戏	158
CD播放器	168
多类型媒体播放器	173
第5章 C++ Builder网络编程	180
建立自己的Web浏览器	180
网络聊天器	189
网络检测	197
获取Web源文件	201
动态建立共享目录	203
POP3邮件接收	209
搜索局域网计算机	219
改进的Ping功能	224
NetBOIS	229
第6章 C++ Builder文件处理	237
文件切割	237
创建快捷方式	242
文件复制	247
文本编辑器	252
显示历史文件	254

文件搜索	258
文件操作	261
第7章 C++ Builder消息API编程	270
利用API创建菜单	270
Tray编程	274
禁止屏保	277
列表风格的打开对话框	279
API获取系统密码	283
硬件检测	297
获取系统目录	305
第8章 其他应用实例	310
创建使用DLL	310
修改IE图片显示	313
调用Word	316
启动检测与关机屏蔽	321
拖放功能实现	323
启动日志	327
COM技术应用	332
第9章 综合实例	342
给水管网优化	342
管网数据库综合应用	352
FTP实例	367

第1章 C++ Builder与控件编程



本章内容

- 圆形、三角形按钮实现
- 用C++ Builder创建组合框
- 用C++ Builder建立动态菜单
- 身份证号识别程序
- 制作闹钟
- 输入法切换
- ListBox自绘
- RichEdit实现上标下标
- 版权窗口的实现
- 不规则窗体的显示
- 动态生产按钮对象
- 保持控件的大小位置
- Memo技巧编程
- ListView头排序
- 本章小结

案例 1

圆形、三角形按钮的实现

由于Windows增加了对窗口区域概念的支持,使得窗口区域就是所定义的窗口看上去的形状,窗口区域作为窗口的实际边界区域,这样窗口区域不仅定义了窗口的可画区域,而且定义了窗口下隐含区域和响应鼠标击键事件的响应窗口区域。

这样，按钮的各种形状是在运行时设置而不是在设计时设置。设计时按钮的形状仍然是长方形或正方形。按钮形状的设计分成两个步骤。

第一步，创建按钮的外形，即定义窗口区域。使用API的区域创建函数就可达到目的。区域创建函数很多，主要有CreateEllipticRgn、CreatePolygonRgn、CreateRectRgn、CreateRoundRectRgn，用于创建不同类型的区域显示。如果需要一个复杂区域形状，那么可以分别使用不同的区域函数创建不同的区域，然后调用CombineRgn API函数把它们组合起来。所以说，不仅可以创建出圆形、三角形的按钮，而且还可以创建出其他各种形状的按钮。

第二步，将区域显示应用到窗口上。一旦使用区域函数创建了一个新的区域显示，就可以使用SetWindowsRgn函数将这个区域应用到窗口上。



窗体的设计

1. 窗体的设计

为了直观显示效果，将窗体的背景色设置为了黑色（Color设置为clBtnText），并将两个按钮的Cursor设置crCross。在窗体中放置两个Button按钮，分别定义为RButton和TButton。界面设计如图1-1所示。

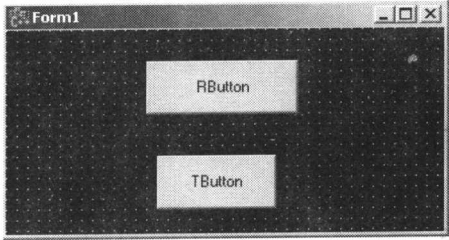


图1-1 窗体设计图



代码实现与详解

2. 头文件及变量设置

其中TrPoint在Unit.h头文件中定义如下：

```
__published: // IDE-managed Components
    TButton *RButton;
    TButton *TButton;
private: // User declarations
    POINT TrPoint[3];
```

3. 窗体代码的具体实现

实现窗口区域的最佳时机是在窗体首次创建时，所以将处理程序放在OnCreate事件中。

```
void __fastcall TForm1::FormCreate(TObject *Sender)
{
    //创建圆形按钮
```

```

HRGN hRgnR = CreateEllipticRgn(0,0,RButton->
Width,RButton->Height);
SetWindowRgn(RButton->Handle,hRgnR,TRUE);
//创建三角形按钮
TrPoint[0].x=TButton->Width/2;
TrPoint[0].y=0;
TrPoint[1].x=0;
TrPoint[1].y=TButton->Height;
TrPoint[2].x=TButton->Width;
TrPoint[2].y=TButton->Height;
HRGN hRgnT = CreatePolygonRgn(TrPoint,3,ALTERNATE);
SetWindowRgn(TButton->Handle,hRgnT,TRUE);
}

```



案例运行效果

在按钮的边界内分别创建了一个圆形区域和一个三角形区域，然后作为新的窗口区域把它赋给各自的按钮。在创建圆形按钮时使用了椭圆形区域创建函数，要使按钮为圆形，在设计时该按钮应设计为正方形。另外区域是以按钮坐标的形式定义的，而不是用屏幕坐标。（0，0）点是按钮的最左上角，而（宽，高）点是按钮的最右下角。程序演示结果如图1-2所示。

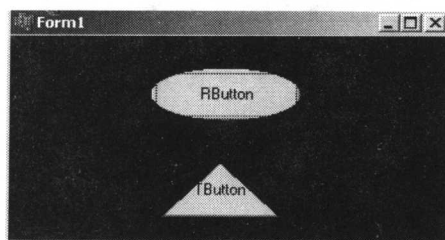


图1-2 程序运行图

应该注意的是，一旦将区域句柄赋给按钮，就不能再对区域句柄作任何操作。如果将区域句柄赋给按钮后，修改或删除这个句柄将导致程序崩溃。

光标只在圆形和三角形区域内变为`crCross`形状。为了使按钮美观可以使用`BitBtn`或`Speed-Button`按钮，在按钮上附上位图。有关API区域函数的使用请参看`Win32SDK.hlp`帮助文件。

以上方法在Win2000pro和C++ Builder 6.0 Enterprise版中实现。

案例 2

用C++ Builder创建组合框

本文介绍了在Windows 2000环境下，如何利用C++ Builder和Windows的API函数创建满足用户特定需求的ComboBox框，并且通过API函数扩充了C++ Builder标准组件的功能。

在C++ Builder下，其所带的ComboBox控件当内容发生变化时将调用OnChange的ComboBox1Change事件句柄。这时事件句柄将向系统发送一条基于ComboBox的CB_FINDSTRING消息，在消息中传递编辑字符串内容。该消息的Send Message返回值是下拉框中第一个匹配条目的Index值，如果在下拉列表框中找到相匹配的条目，Comobox1Change句柄将发送一条基于ComboBox的CB_SETEditSel消息，该消息的参数确定从插入点到字符串结尾的部分将被选择。结果是被选中的部分作为消息搜索结果显示在Combo的Text值中。



窗体的设计

1. 窗体的设计

在进入C++ Builder 6.0以后，新建工程文件，在窗体上画一个Cmobox1，在Items中输入anlantic faloncs、china、japanese、amercia等字符串。界面设计比较简单，如图1-3所示。

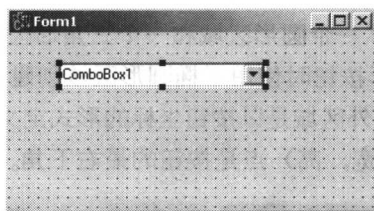


图1-3 窗体设计图



代码实现与详解

2. 头文件及变量设置

在头文件中增加如下变量：

```
Private:WORD LastKey;
```

3. 窗体代码的具体实现

创建ComboBox1的OnChange句柄，并填入如下代码：

```
void _fastcall TForm1::ComboBox1Change(TObject * Sender)
{
    String value = ComboBox1->Text ;
    //如果用户输入的是Delete键，搜索不进行
    if (lastkey==VK_DELETE)
    {
        lastkey = 0 ;
        return ;
    }
    lastkey = 0 ;
    //如果用户的光标在输入字符串的中间，搜索不进行
```

```

if(ComboBox1->SelStart!=value.Length())          return ;
//在下拉列表框中寻找与用户输入字符串相匹配的字符串的索引值
int index=SendMessage (ComboBox1->Handle, CB_FINDSTRING, -1,
(LPARAM) value.c_str ());
if (index >= 0)//如果索引值>0;
{
    ComboBox1->ItemIndex = index ;
//发送CB_SETEDITSEL消息
    SendMessage(ComboBox1->Handle,CB_SETEDITSEL,0, MAKELPARAM
(value.Length (), -1));
}
}

```

创建ComboBox1的KeyDown句柄，并填入如下代码：

```

void _fastcall TForm1::ComboBox1KeyDown(TObject * Sender, WORD * Key,
TShiftState Shift)
{
    lastkey = Key ;//保存最后的键
}

```



案例运行效果

编译运行。当用户在输入框中输入Chi时，便出现China整个字符串。演示结果如图1-4所示。

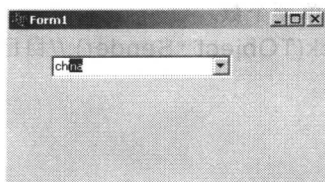


图1-4 程序演示

以上方法在Win2000pro和C++ Builder 6.0 Enterprise版中实现，有时想让控件实现一些不大可能的功能，试试Windows的API函数，或许它可以让你梦想成真。

案 例 3

C++ Builder动态建立菜单

在Windows 2000环境下，利用C++ Builder建立静态的菜单对我们来说是非常容易的，但是如果在我们的软件中能够根据需要在实时地产生动态的菜单，我相信会使我们的软件更加方便实用。

关于动态菜单的建立具体代码实现如下。



窗体的设计

1. 窗体的设计

新建一个工程文件“File | New Application”。然后在Form1上面建立一个PopupMenu1, 一个Button1, 一个RichEdit1。界面设计如图1-5所示。

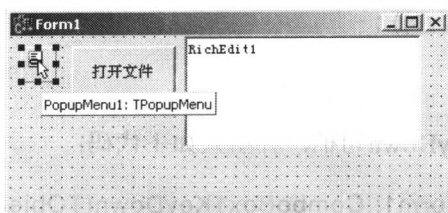


图1-5 窗体设计图



代码的实现与详解

2. 头文件及变量设置

在头文Unit1.h中加入以下内容

```
private: // User declarations
POINT MousePos; //保存鼠标的位置
int PopCount; //保存菜单项个数
void __fastcall MyClick(TObject *Sender); //自己定义的单击事件
```

3. 窗体代码的具体实现

控件事件的具体代码如下:

```
void __fastcall TForm1::Button1Click(TObject *Sender)
{
    for(int i=PopCount-1;i>=0;i--) //如果PopupMenu1里面有菜单项
        PopupMenu1->Items->Delete(i); //就删除, 以免重复建立
    TSearchRec Sr; //用来反回文件名
    TMenuItem *NewItem;
    GetCursorPos(&MousePos); //把当前的鼠标位置保存在MousePos里
    int MenuNameLength; //菜单名的长度
    if(FindFirst("d:\\ *.txt",0,Sr)==0)
    {
        //取得第一个符合条件的文件名; TXT文件的位置, 可自己设定;
        NewItem=new TMenuItem(PopupMenu1); file://建立菜单
        MenuNameLength=Sr.Name.Length(); //文件名的长度
        Sr.Name.SetLength(MenuNameLength-4); //去除 ".TXT"
        NewItem->Caption=Sr.Name; file://新建菜单的名字
        PopupMenu1->Items->Add(NewItem); //把新建菜单项加到PopupMenu1里
```

```

NewItem->OnClick=MyClick; //把自己定义的MyClick函数复给新菜单的OnClick事件
//查找下一个符合条件的文件名，直到完
while(FindNext(Sr)==0)
{
    NewItem=new TMenuItem(PopupMenu1);
    MenuNameLength=Sr.Name.Length();
    Sr.Name.SetLength(MenuNameLength-4);
    NewItem->Caption=Sr.Name;
    PopupMenu1->Items->Add(NewItem);
    NewItem->OnClick=MyClick;
}
FindClose(Sr); //结束查找
}
PopCount=PopupMenu1->Items->Count; //PopupMenu1下的菜单项个数保存
PopupMenu1->Popup(MousePos.x,MousePos.y); //在当前位置弹出PopupMenu1菜单
}
{
    //下面的代码主要是把选定的文件显示在RichEdit1上
    Form1->Caption=((TMenuItem*)Sender)->Caption;
    RichEdit1->PlainText=true; //设为true，表示纯文本文件
    RichEdit1->Lines->LoadFromFile("D:\\\" : +((TMenuItem*)Sender)
    ->Caption+".txt");
}
void __fastcall TForm1::FormCreate(TObject *Sender)
{
    PopCount=0; //初始化
}

```



案例运行效果

上面的程序编译之后，单击“打开文件”，就会弹出PopupMenu1，如图1-6所示。

通过单击PopupMenu1下面的菜单项就会在RichEdit1上面显示文件的内容。结果演示如图1-7所示。

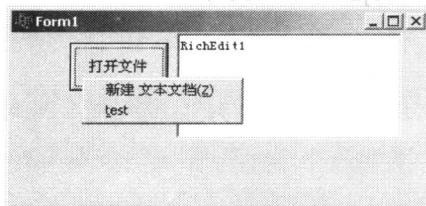


图1-6 运行效果图

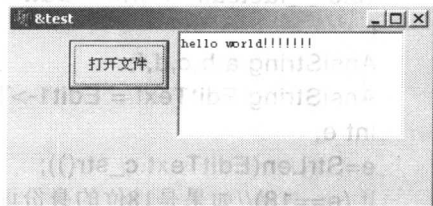


图1-7 打开文件

这个程序很简单，主要是说明如何动态建立菜单及如何响应新建菜单的事件，也可应用到其他上面，如TMenuMain等。

以上方法在Win2000pro和C++ Builder 6.0 Enterprise版中实现。

案例 4

身份证号识别程序

众所周知，我们的身份证号码中包含了我们每个人的基本信息，包括出生年月日和性别，那么如何让软件自己来处理每个人的身份证号码，从中得到有用的信息呢，我们就动手做一个小型的身份证验证程序。

关于身份证验证程序的具体代码实现如下。



窗体的设计

1. 窗体的设计

新建一个工程文件“File | New Application”。然后在Form1上面建立一个输入框Edit1和两个标签Label1、Label2以及一个命令按钮Button1。界面如图1-8所示。

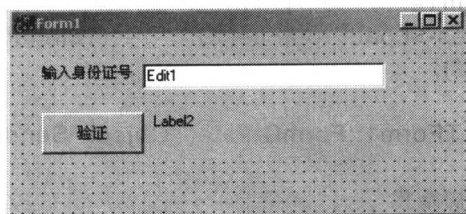


图1-8 窗体设计图



代码实现与详解

2. 窗体代码的具体实现

实现“验证”按钮的OnClick事件：

```
void __fastcall TForm1::Button1Click(TObject *Sender)
{
    AnsiString a,b,c,d,f;
    AnsiString EditText = Edit1->Text;
    int e;
    e=StrLen(EditText.c_str());
    if (e==18)//如果是18位的身份证号
    {
        Edit1->SelStart=6;//从第6位开始
        Edit1->SelLength=4;//取4位数字
        a=Edit1->SelText;//取得出生年的时间
        Edit1->SelStart=10;//从第10位开始
        Edit1->SelLength=2;//取2位数字
```



```

b=Edit1->SelText;//取得出生月的时间
Edit1->SelStart=12;//从第12位开始
Edit1->SelLength=2;//取2位数字
c=Edit1->SelText;//取得出生日的的时间
Edit1->SelStart=16;//从的16位开始
Edit1->SelLength=1;//取1位数字
d=Edit1->SelText;//取得性别的代码
if((StrToInt(d)%2)==1)
{
f="男"; //奇数表示男性
}
else
{
f="女"; //偶数表示女性
}
Label2->Caption="你是"+a+"年"+b+"月"+c+"日生人！"+f+"性！"+
Trim(IntToStr(e))+"位证号！";
return;
}
if (e==15)//对于15老身份证号码，以下代码功能同上
{
Edit1->SelStart=6;
Edit1->SelLength=2;
a="19"+Edit1->SelText;
Edit1->SelStart=8;
Edit1->SelLength=2;
b=Edit1->SelText;
Edit1->SelStart=10;
Edit1->SelLength=2;
c=Edit1->SelText;
Edit1->SelStart=14;
Edit1->SelLength=1;
d=Edit1->SelText;
if((StrToInt(d)%2)==1)
{
f="男";
}
else
{
f="女";
}
Label2->Caption="你是"+a+"年"+b+"月"+c+"日生人！"+f+"性！"+Trim(IntToStr(e))+
"位证号！";
return;
}
ShowMessage("输入的身份证号不正确!");
}

```