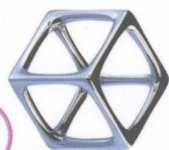


源代码免费下载



商业开发

代码库系列

C++ Builder

案例开发集锦

(第二版)

冯 妍 黄铝文 马 煜 编著
冯 闯 康祥顺 审校



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

C++ Builder

案例开发集锦

(第二版)

本书融合了作者多年的编程经验和从事计算机教学的体会，在重基础能力、勇于创新上下功夫，力求做到循序渐进，通俗易懂。本书以应用为目的，选取有代表性、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、网络等方面的具体应用，突出C++ Builder 6.0的特点和功能，让读者在案例中学习编程思想和编程技巧。

本书的特点包括：

- 案例丰富多样、涵盖了C++ Builder 6.0许多的技术门类和知识要点。
- 案例具有较强的代表性、实用性和趣味性。
- 对每个案例的制作要点进行了介绍，使读者能够举一反三。
- 源代码免费下载，并且对源代码进行了相应的解释，以便读者学习。

商业开发代码库系列丛书：

Java案例开发集锦
Java案例开发集锦（第二版）
JSP案例开发集锦
JSP案例开发集锦（第二版）
Visual FoxPro案例开发集锦
Visual FoxPro案例开发集锦（第二版）
Visual Basic案例开发集锦
Visual Basic案例开发集锦（第二版）
Visual Basic.NET案例开发集锦
Visual Basic.NET案例开发集锦（第二版）
C++ Builder案例开发集锦
C++ Builder案例开发集锦（第二版）
Visual C++案例开发集锦
Visual C++案例开发集锦（第二版）
Visual C++.NET案例开发集锦
Visual C++.NET案例开发集锦（第二版）
Visual C#.NET案例开发集锦
Visual C#.NET案例开发集锦（第二版）
ASP案例开发集锦
ASP案例开发集锦（第二版）
ASP.NET案例开发集锦
ASP.NET案例开发集锦（第二版）
Delphi案例开发集锦
Delphi案例开发集锦（第二版）
PowerBuilder案例开发集锦
PowerBuilder案例开发集锦（第二版）



责任编辑：李莹

本书贴有激光防伪标志，凡没有防伪标志者，属盗版图书。



ISBN 978-7-121-05961-2



9 787121 059612 >

定价：45.00元

TP312/1639=2

2008

商业开发代码库系列

C++ Builder案例开发集锦 (第二版)

冯 妍 黄铝文 马 煜 编著

冯 闯 康祥顺 审校

电子工业出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

本书以C++ Builder 6.0为蓝本,从基础入手,通过大量的案例,系统全面地介绍了C++ Builder 6.0中各个组件/控件的使用方法,以及实际应用系统开发的基本步骤和实现手段。在内容编排上,采取从易到难,循序渐进的原则,所选案例都具有较强的代表性、实用性和趣味性,能激发读者对C++ Builder 6.0的学习热情并感到学有所用,同时,享受C++ Builder 6.0所带来的快乐。

本书内容翔实,案例丰富,条理清晰,注重理论联系实际,适合初学者和已经有一定开发经验的人员作为自学教材或参考书。建议初学者可按照章节从易到难逐步深入学习,对于已经有C++ Builder开发经验的读者,可以直接阅读自己感兴趣的章节。

(附二策)

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/冯妍,黄铝文,马煜编著.—2版—北京:电子工业出版社,2008.4
(商业开发代码库系列)

ISBN 978-7-121-05961-2

I. C… II. ①冯… ②黄… ③马… III. C语言—程序设计 IV. TP312

中国版本图书馆CIP数据核字(2008)第016040号

责任编辑:李莹

印刷:北京天竺颖华印刷厂

装订:三河市金马印装有限公司

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

开本:787×1092 1/16 印张:27.375 字数:700千字

印次:2008年4月第1次印刷

定价:45.00元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至zltz@phei.com.cn,盗版侵权举报请发邮件至dbqq@phei.com.cn。

服务热线:(010) 88258888。

前 言

随着计算机技术和软件产业的发展，为了提高应用程序的高效性和统一性，尽量应用已有的编程经验，把主要的精力放在应用程序的开发上，人们提出了可视化编程技术，Borland公司开发的C++ Builder就是其中具有代表性的可视化开发工具。

C++ Builder是Borland公司在成功推出Delphi之后，又推出的一个基于C++语言的全新的可视化编程环境。它的基础语言为面向对象的C++语言，它继承了C++语言的优点，如代码结构清晰、可读性好和代码执行效率高等，并且是唯一提供真正基于组件的拖放式编程的RAD（快速应用程序），所以C++ Builder一经推出，便迅速得到用户的好评。如同大多数计算机软件一样，Borland公司一直在试图完善这个令人刮目相看的开发软件，所以C++ Builder一直在人们的期待中更新版本。新版的C++ Builder不仅保持了良好的向下兼容性，而且针对软件功能和编译核心都有了不同程度的提高，如ActiveX控件开发应用、Internet应用程序和同步安全的多进程等，从而使得C++ Builder的功能不断强大，使用起来也更加方便和灵活，提高了应用程序的开发效率。

C++ Builder采用了客户/服务器结构，融入了大量的Windows组件，从而使得对象的分类更加详细。C++ Builder将可视化开发环境、可视双向开发工具和可视组件库加入到了C++语言中，它将完全的可视化与真正的面向对象和C++的高效性完美结合起来。总之，C++ Builder是开发Windows应用程序最为强大的工具之一，它正在为越来越多的程序员提供最快速的可视化开发平台、最省力的开发工具和支持平台技术最广泛的开发环境。因此C++ Builder受到了很多企业界程序设计人员的青睐。

近年来，市面上的相关案例书籍要么太简单，不能给读者提供切实有用的帮助；要么难度太大，使得大部分读者不知道如何下手学习。为了满足广大读者的愿望，迅速掌握Windows环境下的C++ Builder 6.0编程方法，作者编写了本书。

本书通过精选的案例，从各个方面讲述了如何利用C++ Builder 6.0开发应用程序，涵盖了界面外观、多媒体控制与图像处理、文件处理、数据库、网络与通信、API编程、游戏编程等内容。各章在内容安排上遵循统一的原则，都是通过经典原创的例子重点讲述如何利用C++ Builder 6.0以及API函数完成特定的开发任务。书中内容由浅入深、循序渐进；知识点全面，覆盖了C++ Builder开发的方方面面；理论与实践紧密结合，注重实用性和实际问题的解决方案；结构清晰、层次分明、分类合理，按照案例运行效果与操作、制作要点、步骤详解和小结的顺序进行阐述。本书考虑了不同层次、不同领域的应用需求，精选的案例覆盖了前面所叙述的大部分知识点（数据库、网络、文件等），而且带有很强的实践针对性，并且完全原创、实用、丰富、解释详尽，读者从中可以提炼出许多不为人知的开发技巧和思路。每个案例完整的源代码，读者可以在华信教育资源网下载。

目 录

第1章 初级编程	1
案例1 动态创建窗体	1
案例2 单选与复选按钮	4
案例3 动态生成多个按钮	7
案例4 菜单设计	10
案例5 定时器应用	13
案例6 不规则窗体	17
案例7 身份证号识别程序	21
案例8 查找替换对话框	24
案例9 自制万年历窗口	28
案例10 动态真彩工具栏	31
案例11 XP风格的控件	36
第2章 多媒体与图形图像	39
案例1 烟花效果演示	39
案例2 绘制各种形状的图形	46
案例3 滚动与推拉效果	57
案例4 图像淡入淡出	62
案例5 锐化图像	66
案例6 边缘增强	69
案例7 图像格式转化	71
案例8 自适应图像尺寸的窗口	74
案例9 播放Flash文件	78
案例10 播放多媒体文件	82
案例11 CD播放器	87
案例12 渐变效果文字滚动	96
案例13 火焰文字演示	104
第3章 文件及系统操作	110
案例1 在拷贝、删除文件时显示飞行的文件夹动画	110
案例2 获取文件日期信息	115
案例3 复制文件	118

案例4	图像浏览器	121
案例5	改变鼠标指针的形状	126
案例6	隐藏桌面图标及任务栏	128
案例7	获取系统信息	132
案例8	建立磁盘管理器	138
案例9	确定Windows所处的目录	144
案例10	更改应用程序图标	146
案例11	检测磁盘空间	149
案例12	系统托盘程序	152
案例13	垃圾文件清理器	155
案例14	按键监控	159
案例15	写字板	161
第4章	网络应用	174
案例1	新闻阅读程序	174
案例2	获取本机的IP地址	178
案例3	获取远程主机日期时间	185
案例4	检测网络连通性	192
案例5	FTP下载工具	196
案例6	Web浏览器	207
案例7	收发电子邮件	212
案例8	域名和IP地址的转换	223
案例9	网卡测试	226
案例10	发送和接收数据	231
第5章	数据库应用	236
案例1	利用ODBC连接Excel	236
案例2	建立数据库	239
案例3	查询数据库	242
案例4	SQL查询	244
案例5	主从表连接	246
案例6	将数据库中的数据写入Word文档	250
案例7	设计报表	252
案例8	决策支持控件	256
案例9	ADO编程	259
案例10	多层数据库应用程序	264

第6章 游戏编程	269
案例1 拯救地球	269
案例2 七巧板游戏	278
案例3 九宫游戏	286
案例4 红心24点游戏	296
案例5 幸运52模拟游戏	301
第7章 其他应用案例	311
案例1 修改IE图标显示	311
案例2 禁止屏保	314
案例3 利用API获取系统密码	316
案例4 创建使用DLL	327
案例5 调用Word	330
案例6 sniff网络嗅探器	334
第8章 综合案例	348
案例1 多人聊天系统	348
案例2 画图板	362
案例3 学生信息查询系统	383
案例4 Windows的屏幕保护程序系统	391
案例5 网络五子棋游戏系统	411



第1章

初级编程

案例1 动态创建窗体

案例运行效果与操作

许多应用程序在运行时，首先要求用户输入用户名及口令，只有通过身份验证才可以进入，例如银行的终端。而登录验证模块只是在程序一开始时才会用到，用户通过验证后的任何操作都不会涉及该模块。所以对应该模块的窗体应当被动态创建，通过验证后立即被销毁。本例通过给主程序添加一个登录验证模块来介绍如何动态创建窗体。案例运行效果如图1-1所示，当通过验证后，显示“Hello”消息框，单击“确定”按钮后，“登录校验”窗体将销毁。



图1-1 案例运行效果

制作要点

什么是动态创建窗体？在C++ Builder中，应用程序所用到的窗体按其被创建的方式可以分为两种，一种叫自动创建的窗体，另一种叫接受探访的窗体。其中前者在程序运行时被自动创建，直到程序结束时被销毁，而后的创建及销毁由设计者手动添加代码实现。默认情况下，窗体都是被自动创建的，这样当应用程序涉及到多个窗体时，程序运行会消耗系统大量内存。采用动态创建方式，就是在应用程序需要的时候创建，用完后立即销毁，避免了自动创建窗体一直占用系统内存的弊端。动态创建窗体在实际编程中是经常用到的技术。

在C++ Builder中，每个窗体控件对应一个由TForm派生的类。当设计一个窗体时，C++ Builder就生成了对应的类。知道这一原理后，就不难实现动态创建窗体。在需要使用窗体的时候，实现对应该窗体的类（即创建该类的一个对象），然后调用它的ShowModal()或Show()方法显示窗体，最后在使用完后使用delete语句销毁该对象。

说了这么多，读者不禁要问，到底窗体是如何被自动创建的呢？其实很简单，通过“项目”菜单下的“查看源代码”命令打开一个.cpp文件，其中包含类似于下面的一段代码：

```
#include <vcl.h>
#pragma hdrstop
//-----
USEFORM("Unit1.cpp", Form1);
USEFORM("Unit2.cpp", Form2);
//-----
WINAPI WinMain(HINSTANCE, HINSTANCE, LPSTR, int)
{
```

```

    {
        Application->Initialize();
        Application->CreateForm(__classid(TForm1), &Form1);
        Application->CreateForm(__classid(TForm2), &Form2);
        Application->Run();
    }
    catch (Exception &exception)
    {
        Application->ShowException(&exception);
    }
    catch (...)
    {
        try
        {
            throw Exception("");
        }
        catch (Exception &exception)
        {
            Application->ShowException(&exception);
        }
    }
    return 0;
}

```

很明显, C++ Builder在应用程序主函数WinMain中调用Application的CreateForm()方法, 自动创建了设计的窗体。要设计一个动态创建的窗体, 可以在设计好窗体后单击“项目”菜单下的“选项”命令或者按快捷键Ctrl+Shift+F11, 打开“项目选项”对话框, 将其移至右边“接受探访的窗体”列表框内, 如图1-2所示。

步骤详解

◆新建项目

启动C++ Builder, 新建一项目, 类型为Application。可以通过菜单“文件”►“新建”►“应用程序”建立, 也可以单击工具栏上的 (新建按钮) 按钮实现。

◆设计主窗体

可以看到, C++ Builder自动生成了一个窗体Form1。修改Form1的Caption属性为“主窗口”, 在其上添加一标签, 效果如图1-3所示。

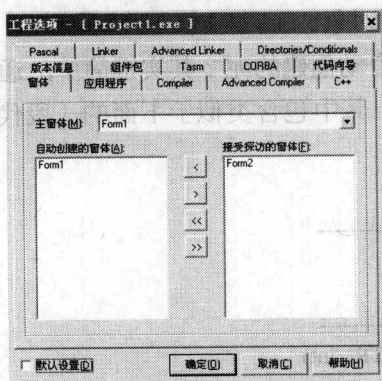


图1-2 “项目选项”对话框

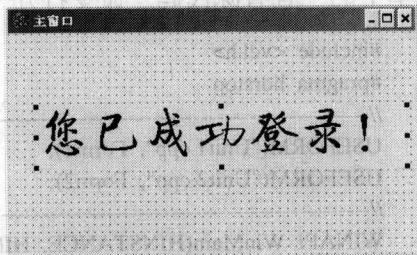


图1-3 程序主界面设计效果

◆添加登录窗体

通过菜单“文件”►“新建”►“窗体”新建一窗体，C++ Builder自动将其命名为Form2。修改Form2的Caption属性为“登录校验”，并在其上添加如下控件。

- Label1 修改其Caption属性为“用户名：”。
- Label2 修改其Caption属性为“密码：”。
- Edit1 修改其Text属性为空，Name属性设为“user”。
- Edit2 修改其Text属性为空，Name属性设为“pswd”，PasswordChar属性设为“*”。
- Button1 修改其Caption属性为“登录”。
- Button2 修改其Caption属性为“退出”。

效果如图1-4所示。

◆添加消息相应函数

通过“项目选项”对话框设置Form2为“接受探访的窗体”，Form1为主窗体，自动创建，如图1-1所示。这样，程序执行后自动创建并显示Form1，为了能达到用户登录校验的目的，需要让Form2先于Form1显示，实现的方法如下。

(1) 选中Form1，在对象查看器的“事件”选项卡中双击OnCreate右边的单元格，给Form1添加消息响应函数onCreate(TObject *Sender)。详细代码如下：

```
void __fastcall TForm1::onCreate(TObject *Sender)
{
    TForm2 *form2 = new TForm2(this); //实例化窗体Form2对应的类TForm2
    form2->ShowModal(); //以模态方式显示窗体
    if(!form2->isLogin())
    { //没有通过验证
        Application->ShowMainForm=false; //不显示主窗体
        PostQuitMessage(0); //退出应用程序
    }
    delete form2; //销毁对象
}
```

(2) 为了完成校验功能，还需为Form2添加相应的函数。首先在Unit.cpp中为Form2对应的类TForm2添加一Boolean型私有成员变量login，用来记录是否通过登录验证，在构造函数里初始化其为false，并添加获取该成员值的方法isLogin()。然后选中Form2，双击“登录”按钮，添加消息响应函数，代码如下：

```
void __fastcall TForm2::Button1Click(TObject *Sender)
{
    if(user->Text!="USER" || pswd->Text!="pswd"); //此处设定用户/密码为USER/pswd
    MessageDlg("用户名或密码错！", mtWarning, TMsgDlgButtons()<<mbYes, 0);
    else
    {
        login = true;
        MessageBox(NULL,"欢迎您!", "Hello!", MB_OK);
        this->Close();
    }
}
```

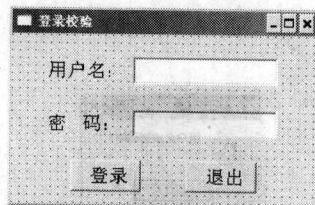


图1-4 登录窗体设计效果

(3) 双击“退出”按钮，添加消息响应函数，代码如下：

```
void __fastcall TForm2::Button2Click(TObject *Sender)
{
    this->Close();
}
```

◆ 调试运行

保存项目文件。编译执行程序后，首先显示“登录校验”对话框，当用户名或密码输入错时，弹出如图1-5所示的消息框，如果选择退出“登录校验”对话框，那也就退出了应用程序。当输入正确时，弹出如图1-6所示的消息框，单击“确定”按钮后，显示主窗口如图1-7所示。

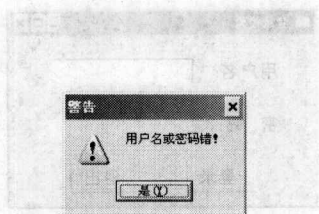


图1-5 登录失败消息框



图1-6 登录成功消息框

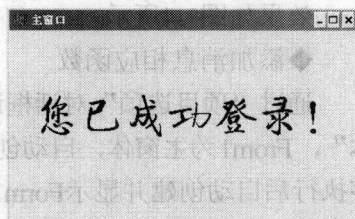


图1-7 主窗口

小结

此案例通过一个简单的程序，主要介绍了动态创建窗体的概念以及在C++ Builer 6.0环境下如何动态创建窗体，同时，应用到了不同类型的消息框。消息框在应用程序设计中经常用到，常常用来提示给用户一些信息，而它的另一个用途是，作为编程人员调试程序的一种手段。

案例2 单选与复选按钮

案例运行效果与操作

图1-8 案例运行效果

通过窗体与用户交互是应用程序中常用到的方法，而这些窗体中往往包含一些单选和复选按钮，以此通过用户鼠标单击选择的方式获取信息。这种方式操作简单，界面友好，因而经常用到。此案例将设计一个调查大学生就业意向的程序，主要调查的信息包括：性别、生源、专业、愿意从事的职业、就业地域意向等。程序的主界面是一个调查窗体，当用户填写完调查单并单击“提交”按钮后，应用程序获取调查结果并以消息框的形式显示出来。其中，调查窗体就大量用到了单选及复选按钮。案例运行效果如图1-8所示。

制作要点

单选与复选按钮属于选择类控件。其中单选按钮允许从多个选项中选择一项，各选项之间的关系是互斥的。在实际应用时，常将多个单选按钮组合成组，构成一个选项组，这就会用到下面将要介绍的另一控件——单选按钮组。复选按钮又叫复选框，它的功能是独立的，多个复选框之间互不影响，允许同时选中多个复选框。在应用中，为了使程序逻辑清晰，常将多个复选框组合在一组，这会用到一个称为GroupBox的控件。下面介绍这几个控件常用的属性、方法和事件。

◆ 单选按钮

- Caption属性** 显示标题，说明单选按钮的功能。
- Alignment属性** 设置单选按钮标题的显示位置。属性值**taRightJustify**表示标题显示在单选按钮右边，属性值**taLeftJustify**表示标题显示在单选按钮左边。该属性的默认值为**taRightJustify**。
- Checked属性** 标记单选按钮是否被选中。选中时该属性的值为**true**，否则为**false**。
- OnClick事件** 当单选按钮被鼠标单击时触发该事件。

◆ 复选按钮

Caption、Alignment和Checked属性的作用与单选按钮的相同。

AllowGrayed属性 该属性说明复选框是否能呈现灰色状态。其值为**true**时，复选框就共有三种状态：**checked**（选中）、**unchecked**（未选中）、**grayed**（灰色）。其值为**false**时，复选框仅有**checked**（选中）和**unchecked**（未选中）两种状态。

State属性 该属性显示复选框处于哪种状态。值**cbUnchecked**表示用户未选中，值**cbChecked**表示选中，值**cbGrayed**表示处于灰色状态。

◆ RadioGroup控件及GroupBox控件

这两个控件相当于单选按钮与复选按钮的父级控件。通过RadioGroup控件的Items属性，可以设定其所包含的单选按钮项。GroupBox控件可以将复选按钮组合在一起，方法是将复选按钮绘制在GroupBox控件中。此外，GroupBox控件也可以组合单选按钮，不过一般不通过此方式组合，因为RadioGroup就是一个专门的单选按钮组。这两个控件都有一个属性Controls，通过它可以获得控件中的单选按钮组或复选按钮组的数组指针，通过此指针可以访问其上的任一单选或复选按钮。

步骤详解

◆ 新建项目

启动C++ Builder，新建一项目，类型为Application。

◆ 设计程序界面

修改C++ Builder自动生成的窗体Form1的Caption属性为“就业意向调查”。在其上添加一标签，内容为“大学生就业意向调查”。

(1) 添加RadioGroup控件

通过Component Palette工具栏的Standard工具箱中的单选按钮组工具，在窗体中添加一个RadioGroup控件，修改其Caption属性为“性别”、Name属性为“Sex”，Coloms属性为“2”，并为其添加单选按钮。方法是，选中该控件，在对象查看器中单击Items属性右边带有三个小点的按钮，打开“列表项编辑器”对话框，如图1-9所示，添加两行内容，单击“确定”按钮。用同样的方法继续添加两个RadioGroup控件，Caption属性分别为“来源”、“专业类别”，对应的Name属性分别为“From”、“Major”，Coloms属性分别为“1”、“2”，并为其添加单选按钮，效果如图1-10所示。

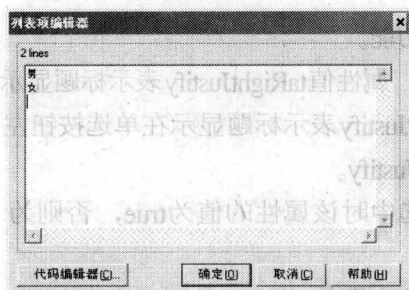


图1-9 单选按钮组列表编辑器

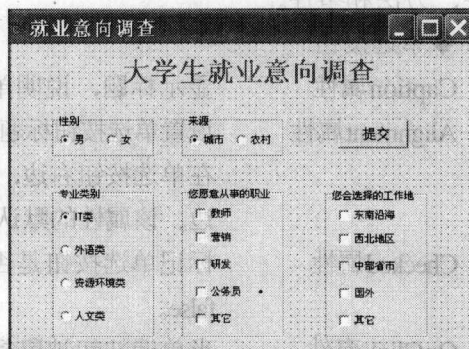


图1-10 程序主界面设计效果

(2) 添加GroupBox控件

通过Component Palette工具栏的Standard工具箱中的GroupBox工具，在窗体中添加一个GroupBox控件，修改其Caption属性为“您愿意从事的职业”、Name属性为“Metier”。然后通过Component Palette工具栏的Standard工具箱中的复选按钮工具在其上添加5个复选框，Caption属性依次为“教师”、“营销”、“研发”、“公务员”、“其它”。同样，再添加一个GroupBox控件，Caption属性为“您会选择的工作地”、Name属性为“Area”，效果如图1-10所示。

(3) 添加按钮

添加一按钮，Caption属性为“提交”，Name属性为“Commit”，双击为其添加OnClick事件的处理方法，详细代码如下：

```
void __fastcall TForm1::CommitClick(TObject *Sender)
{
    //获取被调查者的性别、来源、专业信息并分别保存到变量sex、from、major中
    String sex = "\n性别: ";
    sex += ((TRadioButton *)Sex->Controls[Sex->ItemIndex])>Caption;
    String from = "\n来源: ";
    from += ((TRadioButton *)From->Controls[From->ItemIndex])>Caption;
    String major = "\n专业: ";
    major += ((TRadioButton *)Major->Controls[Major->ItemIndex])>Caption;
    //获取就业意向职业并保存到变量metier中
    String metier = "\n职业: ";
    int c = Metier->ControlCount;
    for(int i=0; i<c; i++)
    {
        if(((TCheckBox *)Metier->Controls[i])>Checked)
            metier += ((TCheckBox *)Metier->Controls[i])>Caption + " ";
    }
}
```

//获取就业意向地区并保存到变量area中

String area = "\n地区: ";

c = Area->ControlCount;

for(int i=0; i<c; i++)

{
if(((TCheckBox *)Area->Controls[i])->Checked)

area += ((TCheckBox *)Area->Controls[i])->Caption + " ";

}

String answer = "调查结果:\n" + sex + from + major + metier + area;

ShowMessage(answer); //显示调查结果

}

◆调试运行

保存项目文件，编译运行。填写窗体，单击“提交”按钮，程序运行结果如图1-11所示。

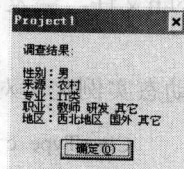


图1-11 调查结果

小结

单选按钮组RadioGroup控件的功能相当于多个单选按钮的综合，具备一些特有的属性。如Colomns属性，用于设置控件中单选按钮排放时的列数；ItemIndex属性，用于返回当前被选中的单选按钮的序号。RadioGroup控件中的单选按钮从上到下其编号从0开始计数，-1表示没有按钮被选中。

与RadioGroup控件不同，GroupBox控件没有提供ItemIndex属性。但是具备ControlCount属性，可以获得控件内复选框的个数，从而可以通过一个循环遍历其中的所有复选框。

单选按钮和复选框是程序设计中使用时频率非常高的控件，通过它们可以取得与用户很好的交互，根据用户的选择决定程序执行结果。比如可以用于软件的选项菜单，让用户对软件进行配置；又如用于软件注册，获取用户的一些信息等。

案例3 动态生成多个按钮

案例运行效果与操作

要动态生成多个按钮，必须理解其中的机制。在案例“动态创建窗体”中已经介绍了动态创建窗体的机制。与动态创建窗体类似，按钮的动态生成就是在程序的运行中实例化按钮控件对应的类——TButton类，最后将实例化的TButton类对象添加到窗体中。在此案例中，通过编写主窗体中的“生成”按钮的OnClick事件响应函数，根据文本框中的数字，动态生成指定数目的按钮，并添加到主窗体中。当重新单击“生成”按钮时，释放先前生成的按钮所占的资源。案例运行效果如图1-12所示。



图1-12 案例运行效果

制作要点

要深入理解动态生成按钮，必须对C++ Builder窗口程序的运行机制有所了解，同时要建立起面向对象的程序设计概念。在C++ Builder中，所有的控件都对应相关的类，如窗体控件对应TForm类、标签控件对应TLabel类、文本框控件对应TEdit类。父控件可以包含一些子控件。一般在设计窗口程序时会将标签、文本框、按钮等控件绘制在窗体控件当中，这时，窗体就是父控件，而其中的标签、文本框、按钮等控件就是它的子控件。如果这时打开窗体对应的.h文件，会发现窗体类中自动添加了响应成员，这些成员就是窗体中的控件对应的类对象。

动态实例化类对象时，会用到C++语言中的new()方法，其使用格式如下：

```
ClassType c = new ClassType(param1,param2,...,paramn)
```

其中ClassType为类名，c为实例化的对象名。括号中的param1, param2, ..., paramn为构造函数的参数列表，为空时表示默认构造。

步骤详解

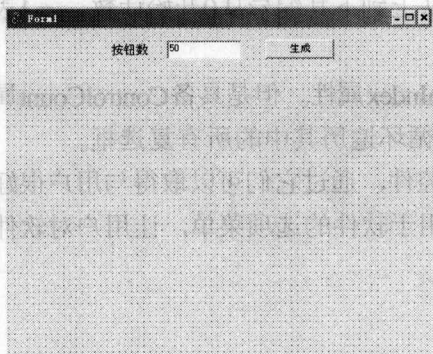


图1-13 程序主界面设计效果

◆新建项目

启动C++ Builder，新建一项目，类型为

◆设计程序界面

在C++ Builder自动生成的窗体Form1上添加一标签，Caption属性为“按钮数”。随后添加一文本框，修改其Name属性为“count”，Text属性为“50”。继续添加一按钮，修改其Caption属性为“生成”。效果如图1-13所示。

◆添加代码

(1) 打开Unit1.h文件，为TForm1类添加如下私有成员及函数：

```
TButton **Buttons;    //按钮类对象指针数组
int BnCount;          //记录已生成按钮的数量
void __fastcall FreeAllButn(void); //用来释放所有已生成的按钮成员函数
void __fastcall OnClick(TObject *Sender); //公用的按钮单击响应函数
```

(2) 添加析构函数：

```
__fastcall ~TForm1();
```

//窗体类TForm1的析构函数

(3) 打开Unit1.cpp文件，为上面的函数添加实现代码如下：

```
//-----
__fastcall TForm1::~TForm1()
{
    FreeAllButn();
}
//-----
```

//释放已生成的所有按钮对象