

Visual C++

数据库实用编程

100 例

```
// CAboutDlg dialog...
class CAboutDlg : public CDialog
{
public:
    CAboutDlg();
// Dialog Data
//{{AFX_DATA(CAboutDlg)
enum { IDD = IDD_ABOUTBOX };
//}}AFX_DATA
// ClassWizard generated virtual function overrides
//{{AFX_VIRTUAL(CAboutDlg)
protected:
virtual void DoDataExchange(CDataExchange* pDX); // DDX
//}}AFX_VIRTUAL
// Implementation
protected:
//{{AFX_MSG(CAboutDlg)
// No message handlers
//}}AFX_MSG
DECLARE_MESSAGE_MAP()
};

CAboutDlg::CAboutDlg() : CDialog(CAboutDlg::IDD)
{
//{{AFX_DATA_INIT(CAboutDlg)
//}}AFX_DATA_INIT
}
```

崔莹 王华军 姚雪峰 编著



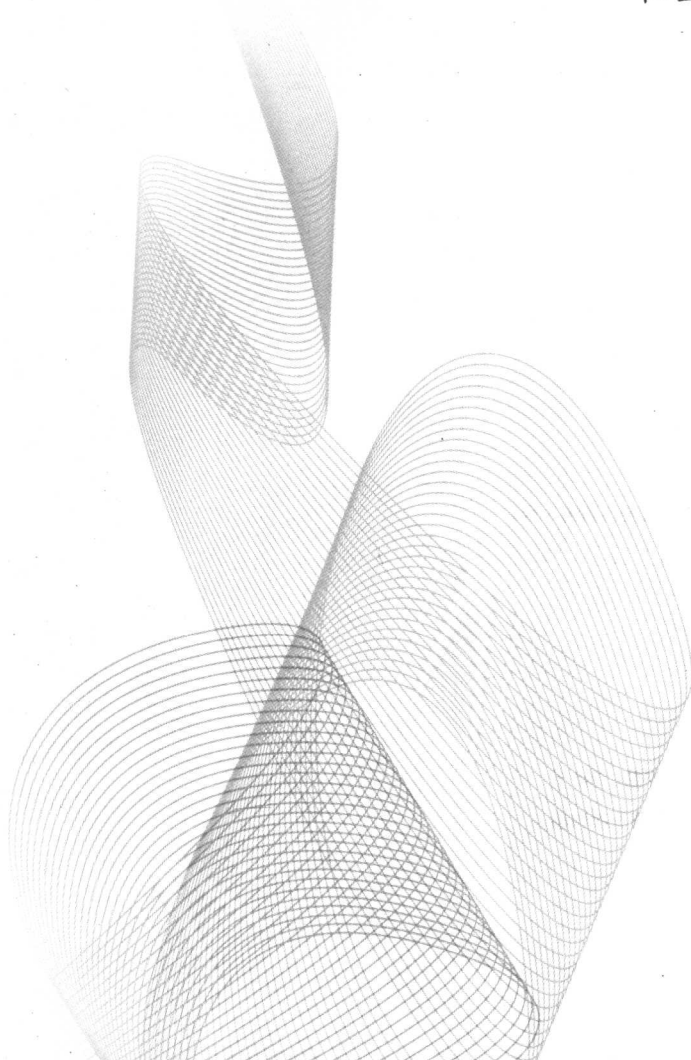
光盘内附书中所有实例源代码

- 囊括100个实例的开发实用手册
- 涉及范围广泛的项目应急宝典
- 学习Visual C++数据库的优秀用书

内容简介

Visual C++ 数据库实用编程 100 例

崔莹 王华军 姚雪峰 编著



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书由浅入深介绍了强大的开发工具 Visual C++ 6.0 与数据库技术。编者精选、设计了 100 个实用经典的实例，以实例带动知识点，使读者可以快速入门并逐步提高。本书实例操作步骤详细，并配以执行结果，直观明了易于理解。本书实例的源程序均可以在随书附带的光盘中找到，读者可以按照操作步骤完成每个实例的制作，并根据自己的喜好和需要修改，做到举一反三。

目前，ODBC 技术是支持最广泛的数据库技术，大量的数据库工程都是使用 ODBC 技术编写的。ADO 技术是微软竭力推广的技术，方便实用，很有前途。鉴于以上原因本书重点介绍了这两种技术。本书内容翔实，是编者多年编程经验的总结，既适合初、中级读者入门与提高又适合高级用户作为参考用书。

图书在版编目 (CIP) 数据

Visual C++数据库实用编程 100 例 / 崔莹, 王华军, 姚雪峰编著. —北京: 中国铁道出版社, 2007. 1

(实用编程百例系列)

ISBN 978-7-113-07707-5

I. V... II. ①崔...②王... ③姚... III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2006) 第 155499 号

书 名: Visual C++数据库实用编程 100 例

作 者: 崔 莹 王华军 姚雪峰

出版发行: 中国铁道出版社 (100054, 北京市宣武区右安门西街 8 号)

策划编辑: 严晓舟 郭毅鹏

责任编辑: 苏 茜 荆 波 姚文娟

封面设计: 薛 为

封面制作: 白 雪

责任校对: 辛 杰

印 刷: 北京市彩桥印刷有限责任公司

开 本: 787×1092 1/16 印张: 25 字数: 592 千

版 本: 2007 年 4 月第 1 版 2007 年 4 月第 1 次印刷

印 数: 1~4 000 册

书 号: ISBN 978-7-113-07707-5/TP·2177

定 价: 40.00 元 (附赠光盘)

版权所有 侵权必究

凡购买铁道版的图书, 如有缺页、倒页、脱页者, 请与本社计算机图书批销部调换。

前 言

当今,世界各国都在切分软件市场的形势下,软件人才依然紧缺。在竞争激烈的市场中,掌握一个功能强大的编程工具无疑是求职的利器,是跨入高收入人群的桥梁。Visual C++ 6.0 是微软推出的开发工具 Visual Studio 6.0 中的一员,其功能十分强大,全面支持面向对象编程技术及组件共享技术,可以大大提高软件开发的速度和软件质量。Visual C++ 6.0 集代码的编、编译、连接和调试于一体,给开发人员提供了一个完整方便的开发界面和许多功能强大的辅助开发工具。

编写简单的界面程序已经不是开发人员的主要任务,几乎所有的中高级开发人员都在进行数据库、网络、通信以及控制方面的开发。本书重点介绍了数据库方面的知识,同时也详细介绍了通信与数据库技术相结合的基本知识。全书共分六个部分,分别为 MFC ODBC 编程、DAO 编程、OLE DB 编程、ADO 编程、数据库的特殊处理以及高级应用,本书重点介绍了 ODBC 技术和 ADO 编程技术以及通信与数据库相结合的技术。

本书的特色是借助于大量的编程实例讲解知识点,使读者在实际编程过程中体会、理解和掌握知识点,同时对涉及到的知识点进行了必要的讲解,避免讲解深奥的道理。这样大大提高了本书的可读性和趣味性,读者每学习一个实例就会有一些提高。

本书虽然是一本以实例为主的书,但本书介绍的知识点由浅入深,是一个有机的整体。本书所介绍的每一个实例都是编者的编程经验总结和结晶。

本书内容全面、新颖,叙述简洁明了。借助于丰富的图形、典型的范例和循序渐进深入浅出的论述方式,力图使读者快速掌握并灵活运用。只要读者能够仔细阅读本书,理解了本书中相关的实例,配合本书实例说明及光盘中的实例代码上机实习,在较短的时间内掌握一种数据库技术是完全可能的。

本书所有的实例的源代码均调试通过,通过本书的介绍读者应该能够轻松的掌握数据库编程技术以及具体编写的方法和技巧。

本书由崔莹担任主编,王华军、姚雪峰担任副主编,另外,谢华锋、刘伟、徐杰、张江涛、于秋生、赵磊、杜江、刘旭、周鸣扬、范翠丽、黄丽娜、郑艳华、谢振华、唐兵、张俊岭、尹建民、冉林仓、李东玉、梁云高、梁斌、李士良、郑砚等人也参加了本书部分内容的编写及素材整理工作,在此一并表示感谢。

由于编者水平有限,加之时间仓促,书中难免有疏漏之处,敬请读者和同行批评指正。

编 者
2006 年 12 月

目 录

第一部分 MFC ODBC 编程

实例 1	注册数据源	2
实例 2	浏览数据库	4
实例 3	使用连接	9
实例 4	添加、删除、修改记录	12
实例 5	捕获异常与处理	16
实例 6	在 ODBC 中应用 DDX 和 RFX	19
实例 7	条件查询	23
实例 8	参数化查询	27
实例 9	条件排序记录集	30
实例 10	单独使用 CRecordset 类	35
实例 11	动态连接表	39
实例 12	动态获得记录集的字段信息	42
实例 13	显示记录集全部记录	46
实例 14	排序并显示记录集全部记录	50
实例 15	使用列表视图显示记录	55
实例 16	在列表视图中删除记录	60
实例 17	在列表视图下排序显示记录	63
实例 18	类的替换及数据库支持	67
实例 19	自动注册数据源	70
实例 20	操作远程数据库	74
实例 21	直接执行 SQL 语句	76
实例 22	多记录集查询	80
实例 23	多记录集删除记录	84
实例 24	新建表	89
实例 25	事务处理	93
实例 26	开发 ODBC 报表 (一)	98
实例 27	开发 ODBC 报表 (二)	102

第二部分 DAO 编程

实例 28	新建应用程序	106
实例 29	管理事务	109
实例 30	打开数据库	112
实例 31	打开记录集	115

实例 32	遍历记录集	118
实例 33	添加记录	122
实例 34	删除记录	126
实例 35	编辑记录	131
实例 36	记录排序	135
实例 37	直接执行 SQL 语句	138
实例 38	创建新表	142
实例 39	综合查询	146
实例 40	存取图像字段	151
实例 41	百分数转化	155
实例 42	获取数据库信息	157
实例 43	直接执行 SQL 语句进行查询	162

第三部分 OLE DB 编程

实例 44	浏览记录	168
实例 45	添加删除记录	171
实例 46	查询记录集	174
实例 47	使用连接	178
实例 48	使用书签	181
实例 49	使用 ATL 模板创建 OLE DB 客户程序	184
实例 50	开发 OLE DB 报表	190

第四部分 ADO 编程

实例 51	创建应用工程	196
实例 52	表的添加与删除	199
实例 53	表的字段操作	203
实例 54	表名的获取	207
实例 55	单列的添加与删除	211
实例 56	遍历记录集	215
实例 57	添加记录	221
实例 58	修改记录	225
实例 59	整列的数据操作	229
实例 60	删除记录	232
实例 61	查询记录	236
实例 62	统计记录	241
实例 63	设置索引	245
实例 64	求域的极值	249
实例 65	求域的均值	253

实例 66	中文大写的自动生成	257
实例 67	时间段检索	264
实例 68	字符匹配检索	268
实例 69	多字符串匹配检索	272
实例 70	数据备份	276
实例 71	基于自定义 SQL 语句的综合查询	280
实例 72	使用 DataGrid 控件开发应用程序	284
实例 73	一个完整的图像处理程序设计过程 (一)	287
实例 74	一个完整的图像处理程序设计过程 (二)	291
实例 75	一个完整的图像处理程序设计过程 (三)	295
实例 76	一个完整的图像处理程序设计过程 (四)	300
实例 77	保存二进制大对象	303
实例 78	读取二进制大对象	308
实例 79	使用树控件示例	310
实例 80	关闭数据库	316

第五部分 数据库的特殊处理

实例 81	添加 SQL Server ODBC 数据源	320
实例 82	新建 SQL Server 数据库工程	322
实例 83	新建表并添加记录	325
实例 84	SQL 常用函数简介	328
实例 85	新建视图	332
实例 86	操作加密数据库	335
实例 87	数据有效性监测	338
实例 88	建立 Excel 数据库并添加记录	343
实例 89	读取 Excel 数据库中的数据	346
实例 90	访问 FoxPro 数据库	350

第六部分 高级应用

实例 91	用户登录模块	353
实例 92	数据库在 AutoCAD 中的应用 (一)	356
实例 93	数据库在 AutoCAD 中的应用 (二)	360
实例 94	串口通信简介	364
实例 95	使用串口发送数据库中的数据	368
实例 96	串口接收数据并保存到数据库	372
实例 97	网络通信发送端简介	375
实例 98	网络通信接收端简介	380
实例 99	通过网络发送数据库中的数据	384
实例 100	通过网络接收数据并保存到数据库	387

第一部分

MFC ODBC 编程

该部分主要讲解如何使用 ODBC 技术访问及处理数据库。ODBC (Open DataBase Connectivity) 即开放式数据互联技术或者说访问不同数据源的软件开发包。在 20 世纪 90 年代早期, 有几家企业在分别开发自己的数据库产品, 并且每个企业的数据库产品都拥有不同的专有接口。要在应用程序中处理数据库, 如果要处理多个的话, 那么就要为各个不同的数据库编写不同的接口代码, 这样在处理数据库的时候, 将会非常麻烦。随着市场上对数据库使用的越来越多, 对数据库的要求也越来越高, 为了解决这个问题, 由微软公司牵头与其他一些公司合作研究一个标准的接口, 这个标准接口可以从不同的数据库获取数据、版本等信息, 当然不同的数据库也可以接收此标准接口发送的数据。由于这个接口是开放的且同其他数据库互联的技术, 于是人们称这个技术为 ODBC。

ODBC 功能异常强大, 开发人员仅仅使用很少的代码就可以实现各种不同的数据库访问。应用程序通过 ODBC 调用 ODBC 驱动程序管理器, ODBC 驱动程序管理器将调用传递给数据库驱动程序, 数据库驱动程序利用 SQL 语言同数据库管理系统进行数据传递处理。目前不但大多数数据库都拥有 ODBC 驱动器, 支持 ODBC 技术, 同时一些开发工具诸如 Visual C++、Visual Basic 以及 PowerBuilder 等也支持 ODBC 技术, 但是它们对 ODBC 支持的能力可能是不同的, 并且也可能不完全兼容, 这一点是值得注意的。

MFC 改进了 ODBC, 原始的 ODBC 是简单的函数即 ODBC API。MFC 封装了 ODBC API, 它提供了一组类。其中, CDatabase 类主要用来连接数据库、控制数据传输、执行 SQL 语句 (如: Insert、Update 和 Delete 等) 和进行事务处理; CRecordset 类主要用来获取记录集, MFC 利用 CRecordView 构成表处理记录集, 一个记录集是一个在内存中形成的内部表, 它包含了对数据库查询的结果。记录集对象的字段数据成员组成了显示和编辑缓冲区, 该缓冲区中保存了已经选择的记录列; CRecordView 类用来显示记录, 它总是与 CRecordset 类紧密相连利用 RFX 和 DDX 自动跟踪记录集中的变化。在对数据库要求不高的程序中应用 CRecoedView 类显示记录有很大优势, 只需要编写很少的代码就可以实现对记录集的浏览和对数据库的更改。但是它也有很大的局限性, 一次只能对一个记录进行操作, 很不方便。在本部分介绍了利用 CListView 类显示记录, 并且代码同数据库分离, 当数据库改变后代码不变就可以处理不同的数据库。

实例 1

注册数据源

实例说明

在本实例中将介绍数据库编程的基本知识，即如何注册数据库，注册数据库是数据库开发的第一步。特别对于不是很复杂的、不需要在多个数据库中经常切换操作的应用程序，应用注册的数据源无疑是最简单直接的方法。

技术要点

- 掌握注册数据源的步骤。
- 了解注册数据源后操作系统的变化，即 VC 是如何根据数据源来操作数据库的。

实现步骤

(1) 依次打开“控制面板”→“管理工具”→“数据源”，将弹出“ODBC 数据源管理器”对话框，如图 1-1 所示。

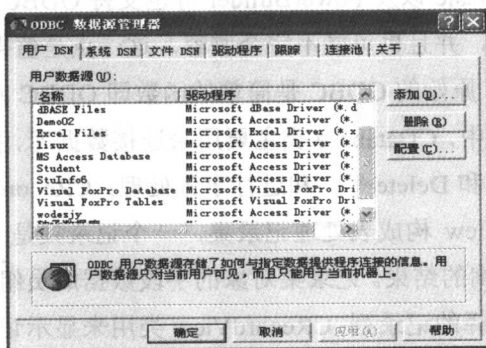


图 1-1 “ODBC 数据源管理器”对话框

(2) 选择“用户 DSN”选项卡，单击“添加”按钮，将会弹出“创建新数据源”对话框，如图 1-2 所示。

(3) 在此对话框中，选择要注册的数据库的驱动程序，假定为 Microsoft Access Driver (*.mdb)，单击“完成”按钮，将弹出“ODBC Microsoft Access 安装”对话框，如图 1-3 所示。



图 1-2 “创建新数据源”对话框

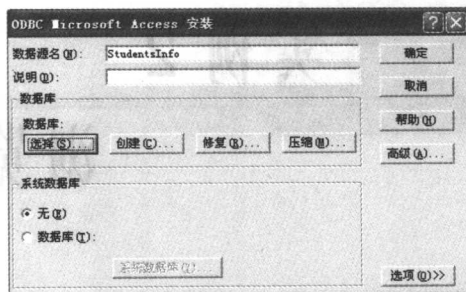


图 1-3 “ODBC Microsoft Access 安装”对话框

(4) 在“数据源名”文本框中输入要注册的数据源的名字，这个名字可以依个人的喜好，填写任意的名字。然后单击“选择”按钮，将弹出“选择数据库”对话框，选择要注册的数据库，如图 1-4 所示。

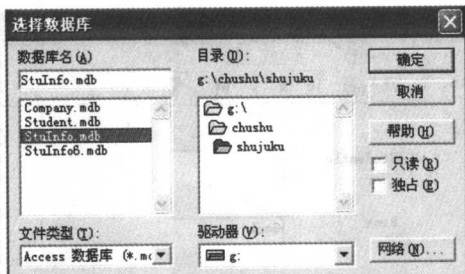


图 1-4 “选择数据库”对话框

(5) 选择要注册的数据库假定为 StuInfo.mdb，然后单击“确定”按钮，注册数据源的工作就算完成了。



归纳注释

- (1) 本实例完全依照注册数据源模板的步骤，一步一步完成了数据源的注册。
- (2) 操作系统有多种数据库，注册数据源后，数据库的字段信息以及驱动程序等信息将反映在系统的注册表中。当在应用程序中要用到数据源的时候，应用程序会根据注册表中反映的这个数据源的数据库信息来完成对数据库的操作。

实例 2

浏览数据库

实例说明

在本实例中将学习怎样浏览数据库中数据表的数据,可以将数据库中的数据逐条浏览。通过按钮控件可以将记录移动到第一条记录、最后一条记录以及上一条记录和下一条记录。程序运行结果如图 2-1 所示。

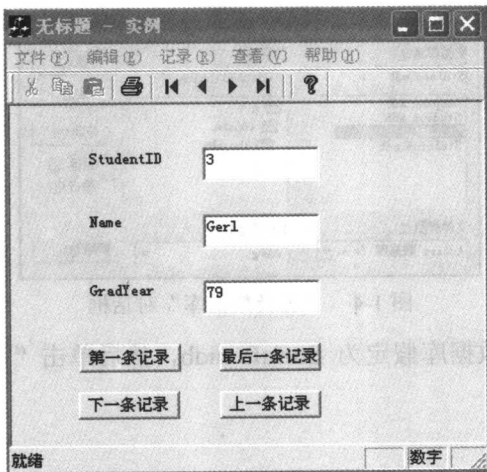


图 2-1 运行结果

技术要点

- 了解类 CRecordset, 掌握并运用此类的操作成员函数, 来完成对数据库的浏览。
- 运用 CRecordset 类的属性成员函数判断数据集的属性。
- 绑定记录集中的记录。

实现步骤

- (1) 注册数据源, 本实例提供一个名为 StuInfo6 的数据源。
- (2) 新建项目。

① 使用 AppWizard 向导创建一个名为“实例”的基于 SDI 的应用程序。在 AppWizard 向导的第 2 步中选择 Database view without file support 选项。然后 Data Source 按钮将变为可用, 单击此按钮将会弹出 Database Options 对话框, 如图 2-2 所示。

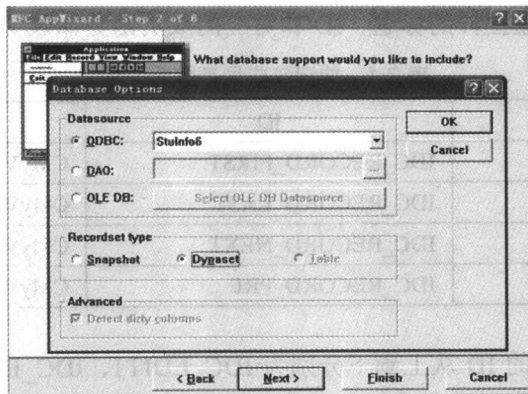


图 2-2 Database Options 对话框

② 在此对话框中选中 ODBC 单选按钮和 Dynaset 单选按钮,单击 OK 按钮,弹出 Select Database Tables 对话框,在此对话框中选中“表 1”单击 OK 按钮,数据源选择完成,如图 2-3 所示。

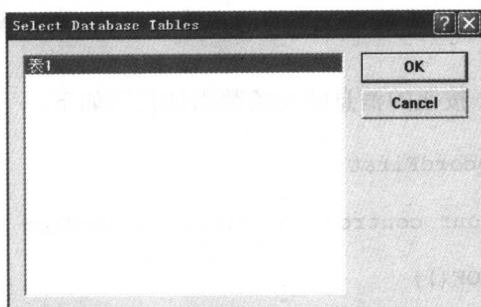


图 2-3 选择数据表

③ 其他的步骤均采用默认设置。

(3) 为应用程序添加控件和代码。

① 在 IDD_MY_FORM 对话框中添加控件按钮,如图 2-4 所示。

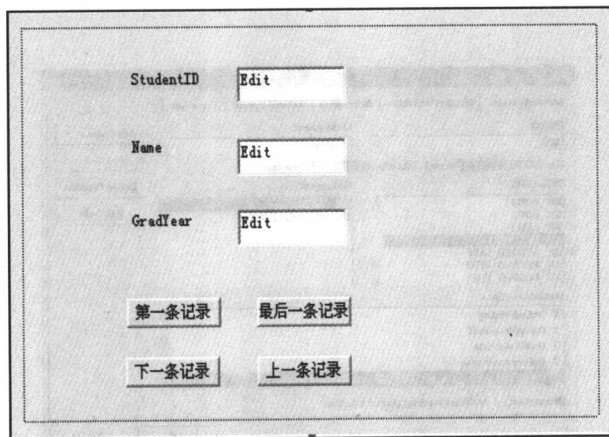


图 2-4 对话框布局

图 2-4 中的 4 个按钮控件的标题、ID 和对应的成员函数如表 2-1 所示。

表 2-1 按钮与成员函数的关联

标 题	ID	对应的成员函数
第一条记录	IDC_RECORD_FIRST	CMyView::OnRecordFirst()
最后一条记录	IDC_RECORD_LAST	CMyView::OnRecordLast()
下一条记录	IDC_RECORD_NEXT	CMyView::OnRecordNext()
上一条记录	IDC_RECORD_PRE	CMyView::OnRecordPre()

图中 3 个编辑控件的 ID 从上到下分别为 IDC_EDIT1、IDC_EDIT2、IDC_EDIT3，对应的记录集的成员变量分别为 m_pSet->m_ID、m_pSet->m_Name、m_pSet->m_GradYear。标题分别为 StudentID、Name、GradYear 的控件为静态控件。

② 选择 View→Class Wizards 命令，弹出 MFC ClassWizard 对话框，如图 2-5 所示。在对话框中的 Class name 下拉列表框中选中 CMyView 类，在 Object IDs 列表框中选中 IDC_RECORD_FIRST 项，在 Messages 列表框中选中 BN_CLICKED 项，然后单击 Add Function 按钮，添加按钮“第一条记录”的消息响应函数。照此添加其他 3 个按钮的消息响应函数。

③ 为“第一条记录”按钮的消息响应函数添加代码如下：

```
//移动到第一条记录
void CMyView::OnRecordFirst()
{
    // TODO: Add your control notification handler code here
    //判断是否为空
    if(m_pSet->IsBOF())
    {
        m_pSet->SetFieldNull(NULL);
    }
    else
    {
        m_pSet->MoveFirst();
        UpdateData(FALSE); //更新对话框窗口
    }
}
```

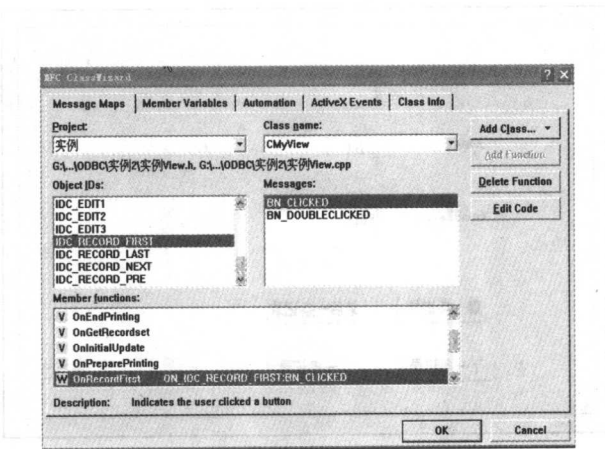


图 2-5 添加成员函数

④ 为“最后一条记录”按钮的消息响应函数添加代码如下:

//移动到最后一条记录

```
void CMyView::OnRecordLast()
{
    // TODO: Add your control notification handler code here
    m_pSet->MoveLast();
    UpdateData(FALSE); //更新对话框窗口
}
```

⑤ 为“下一条记录”按钮的消息响应函数添加代码如下:

//移动到下一条记录

```
void CMyView::OnRecordNext()
{
    // TODO: Add your control notification handler code here
    if(m_pSet->IsBOF())
    {
        m_pSet->SetFieldNull(NULL);
    }
    //判断是否溢出
    if(m_pSet->IsEOF())
    {
        AfxMessageBox("已经到达最后记录");
    }
    else
    {
        m_pSet->MoveNext();
    }
    UpdateData(FALSE);
}
```

⑥ 为“上一条记录”按钮的消息响应函数添加代码如下:

//移动到上一条记录

```
void CMyView::OnRecordPre()
{
    // TODO: Add your control notification handler code here
    if(m_pSet->IsBOF())
    {
        AfxMessageBox("已经到达第一个记录");
    }
    m_pSet->MovePrev();
    UpdateData(FALSE);
}
```

归纳注释

(1) 读者可能会问 `m_pSet` 是什么, 打开“实例 `View.h`”时可以找到“`CMySet* m_pSet;`”这句话, 其实它就是类 `CMySet` 的指针, 而类 `CMySet` 的父类是类 `CRecordset`。这样就应该明白了为什么可以这样使用 `m_pSet` 了。

(2) 对于 `UpdateData()` 函数, 此函数有两个参数分别为 `TRUE` 和 `FALSE`。当为 `TRUE` 时, 其功能为用编辑框中的内容更新成员变量, 当为 `FALSE` 时, 其功能为用成员变量更新

编辑框中的内容。

(3)类 CRecordset 为记录集类,此类是由 MFC 封装的专用于处理数据库记录集类。它拥有一些成员函数处理数据库。在本实例中用到了如下几个函数,它们分别为 MoveFirst()、MoveLast()、MoveNext()和 MovePrev(),其作用分别为移动到第一条记录、移动到最后一记录、移动到下一条记录和移动到上一条记录。

函数 IsEOF()和函数 IsBOF()同样也是 CRecordset 类封装的函数,它们的功能分别为判断数据库是否溢出和是否为空。

实例 3

使用连接

实例说明

本实例将介绍如何在 MFC 应用程序中使用连接。在开发应用程序时经常需要将一个以上的表在视图中显示出来,那么使用 MFC 应用程序向导的表连接是最方便的方法。本实例将根据两个表中学生的学号把两个表连接起来,同时把学生的学号、姓名、毕业时间、家庭住址以及家庭电话在屏幕上显示出来。

本实例的最终结果如图 3-1 所示。

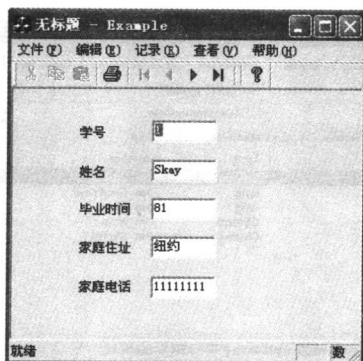


图 3-1 运行结果

技术要点

依据相关的规则连接两个表。

实现步骤

(1) 注册数据源,本实例使用的数据源同上实例。

(2) 为数据源添加一个名为 Basic 的表。

(3) 新建工程。

① 本实例新建一名为 Example 的单文档工程。

② 在 AppWizard 向导的第 2 步中选择 Database view without file support 选项,然后单击 Data Source 按钮。

③ 在 Database Options 对话框中,选中 ODBC 和 Dynaset 单选按钮后选择 ODBC 数据源。打开 Select Database Table 对话框,选择两个表连接在一起,如图 3-2 所示。

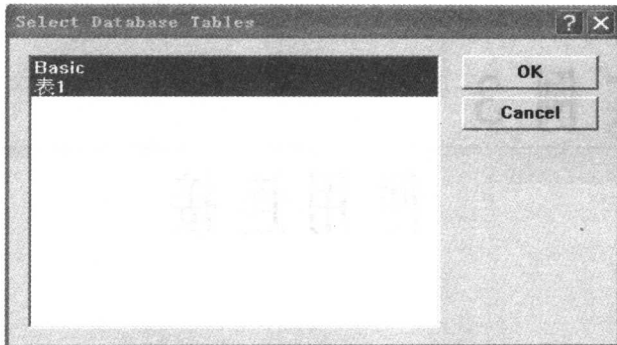


图 3-2 选择数据表

④ 其他步骤默认。

(4) 添加控件。

如图 3-1 所示为工程添加控件。控件的 ID 可以随读者的意愿取容易记忆的名字。

(5) 绑定字段与编辑框。

使用类向导把编辑控件和数据库中的字段绑定在一起，如图 3-3 所示。

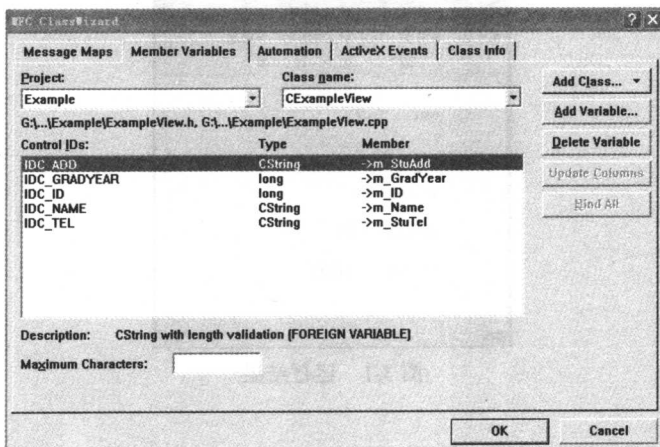


图 3-3 绑定记录

(6) 使用 CRecordset 类的成员变量添加连接两个表的标准。代码如下：

```

CExampleSet::CExampleSet(CDatabase* pdb)
    : CRecordset(pdb)
{
   //{{AFX_FIELD_INIT(CExampleSet)
    m_StuID = 0;
    m_StuAdd = _T("");
    m_StuTel = _T("");
    m_ID = 0;
    m_Name = _T("");
    m_GradYear = 0;
    m_nFields = 6;
    //}}AFX_FIELD_INIT
    
```