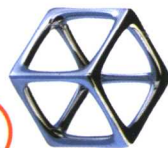


源代码免费下载

商业开发



代码库系列

Visual Basic

案例开发集锦

(第二版)

周 峰 李德路 王 征 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

内 容 简 介

本书使用代表性和实用性强、效果新颖的案例讲解了Visual Basic在数据库、窗体界面、多媒体、图形图像、字符串处理、系统文件、网络、硬件、游戏、API函数及第三方控件等方面的具体应用,基本涵盖了Visual Basic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,在华信教育资源网上可免费下载源代码,读者可在案例练习中体会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

Visual Basic案例开发集锦/周峰,李德路,王征编著. —2版. —北京:电子工业出版社,2008.1

(商业开发代码库系列)

ISBN 978-7-121-05063-3

I. V… II. ①周…②李…③王… III. BASIC语言—程序设计 IV. TP312

中国版本图书馆CIP数据核字(2007)第145802号

责任编辑:戴 新

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:25.25 字数:640千字

印 次:2008年1月第1次印刷

定 价:38.00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zlts@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

随着计算机技术的不断发展，许多功能强大的软件纷纷面世，最新出现的软件都采用了图形界面，操作简单方便，使用灵活。**Visual Basic**就是其中最典型的一种可视化的编程软件。

Visual Basic使用了**Microsoft Windows**图形用户界面的许多先进特性和设计理念，提供了向导自动生成代码，具有属性提示与检查、实时语法检查、拖动技术等内置自动化特性，大大减少了编写代码的工作量，并且提供了非常丰富的控件。这些特点使得**Visual Basic**成为最简捷、最有效的开发**Windows**应用程序的工具。

本书从实际应用入手，利用具有代表性、实用性以及效果新颖的案例讲解了**Visual Basic**在数据库、图形图像、API函数、游戏等方面的应用。本书融入了作者多年实际编程的经验和体会，每一个案例都是精心挑选的，解释详尽，使读者在案例练习中体会并掌握编程思想和编程技巧。

本书结构

本书共分11章，具体章节安排如下：

- ◆ 第1章：通过14个具体案例讲解了**Visual Basic**数据库程序开发的强大功能及应用。
- ◆ 第2章至第5章：讲解了**Visual Basic**在窗体界面、图形图像、多媒体、游戏方面的应用及编程技巧。
- ◆ 第6章至第8章：讲解了**Visual Basic**在系统文件、硬件及API函数方面的应用。
- ◆ 第9章至第10章：讲解了**Visual Basic**在网络和第三方控件及其他方面的具体应用。
- ◆ 第11章：通过3个综合案例讲解了**Visual Basic**开发文件系统、游戏及数据库系统的方法与技巧。

本书特色

本书的特色归纳如下：

- ◆ 经典且效果新颖的实例

第1章至第10章通过大量具有代表性、实用性以及效果新颖的案例讲解了**Visual Basic**在窗体界面、图形图像、多媒体、游戏等方面的具体应用。

◆ 代码注重可读性和实用性

本书编写的实例都是经过反复测试的，如综合案例是在实际开发中经常用到的。本书能够让读者真正地在研究实例的同时，完全掌握Visual Basic编程的精髓。

本书适合的读者

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，对于专业人士也有较高的参考价值。

本书重点介绍Visual Basic编程技术，写作模式是用实例剖析讲解，使得读者能够迅速上手开发具体的应用程序。本书是实例书籍，每个实例都涵盖了多个技术要点，而且注重实际应用，明显高于其他同类书籍。读者在深入研究本书后，能够提高自己的编程水平，达到豁然开朗、柳暗花明又一村的效果。

以下人员对本书的编写提出过宝贵意见并参与了本书的部分资料搜集工作，他们是解翠、孙宁、王荣芳、李岩、周科峰、陈勇、高云、王春玲、韩亚男、陈卓、甄晓阳、于凯、姚国发，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。

由于时间仓促，加之水平有限，书中的缺点和不足之处在所难免，敬请读者批评指正。

为方便读者阅读，若需要本书配套资料，请登录“华信教育资源网”（<http://www.hxedu.com.cn>），在“资源下载”频道的“图书资源”栏目下载。

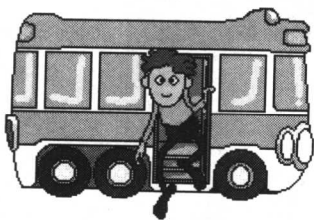
目 录

第1章 Visual Basic与数据库	1
案例1 显示所有数据库表及表中的数据	1
案例2 图表显示数据库中的数据	4
案例3 动态查询数据库中的数据	8
案例4 利用DataGrid控件显示数据库中数据信息	11
案例5 利用Printer对象直接打印数据库中的数据	16
案例6 调用Excel软件实现打印预览功能	18
案例7 动态改变数据库中数据图表	21
案例8 带有图像信息的数据库数据的操作	27
案例9 带有多媒体信息的数据库数据的操作	34
案例10 动态创建访问数据库的DSN	43
案例11 动态创建数据库及数据库表	47
案例12 分页显示数据库表中的记录	50
案例13 动态显示数据库中图片信息	52
案例14 编程实现利用向导把文本文件导入到数据库中	57
本章小结	69
 第2章 Visual Basic与窗体界面	 70
案例1 带有滚动条的窗体	70
案例2 放大窗体	73
案例3 动画窗体	75
案例4 工具箱窗体	77
案例5 奇形位图窗体	81
案例6 MDI窗体	83
案例7 喷枪窗体	90
案例8 三角形动画窗体	92
案例9 工字形窗体	97
案例10 运行时钟窗体	99
案例11 闪烁窗体	101
案例12 中空圆角矩形椭圆窗体	102
案例13 渐变颜色窗体	105
本章小结	108

第3章 Visual Basic与图形图像处理	109
案例1 图像的动态选择与浏览	109
案例2 动态缩放、显示或隐藏图像	111
案例3 形状图形及形状图形填充样式的动态改变	114
案例4 随鼠标移动的彩蝶	117
案例5 动态图像与动态文字	120
案例6 图像电子相册	123
案例7 随机跳动的小球	125
案例8 动态绘制花样图形	128
案例9 高速旋转的立方体	132
案例10 多色渐变多边形	135
案例11 图像淡入与淡出效果	136
本章小结	139
第4章 Visual Basic与多媒体技术	140
案例1 下雨特效	140
案例2 文字动画特效	142
案例3 数字发音机特效	147
案例4 电子琴	151
案例5 枪射击动画特效	157
案例6 Flash动画播放器	160
案例7 VCD播放器	163
案例8 英文翻译朗读程序	164
案例9 火焰特效	166
案例10 图像数字时钟	176
本章小结	178
第5章 Visual Basic与游戏	179
案例1 扫雷游戏	179
案例2 井字形游戏	191
案例3 智力拼图游戏	199
案例4 黑白棋游戏	205
本章小结	218
第6章 Visual Basic与文件	219
案例1 动态创建文件并向文件中输入内容	219
案例2 动态读取文件中的内容	222

案例3 动态打开EXE程序	224
案例4 显示所选文件的属性	226
案例5 记事本文档	230
案例6 文本文件的动态提示效果	233
案例7 图形字符的输出	236
本章小结	240
第7章 Visual Basic与硬件	241
案例1 声卡信息统计	241
案例2 统计键盘信息	245
案例3 光驱的打开与关闭	248
案例4 鼠标移动区域的限定与释放	249
案例5 获得计算机系统信息	253
案例6 计算机可用端口信息统计	255
案例7 硬盘信息的显示与卷标的设定	258
本章小结	260
第8章 Visual Basic与API函数	261
案例1 API函数实现AVI动画鼠标	261
案例2 API函数实现任务栏的显示与隐藏	264
案例3 API函数实现特殊菜单效果	266
案例4 API函数实现动画窗体特效	270
案例5 API函数实现动态提示功能	276
案例6 API函数实现菜单的显示与隐藏	277
本章小结	280
第9章 Visual Basic与网络	281
案例1 创建网页快捷方式	281
案例2 利用WebBrowser控件播放GIF动画	285
案例3 API函数实现超链接功能	287
案例4 网络拨号连接的判断	291
案例5 网络聊天室	293
案例6 API函数实现网络程序的监控	295
案例7 Internet网页浏览器	298
本章小结	301

第10章 Visual Basic第三方控件及其他应用	302
案例1 第三方控件实现窗体渐变色效果	302
案例2 第三方控件实现图像的各种处理	305
案例3 第三方控件实现特殊进度条效果	308
案例4 计算器	316
案例5 简易画板	319
案例6 菜单的动态增加与删除	323
案例7 日历表	325
案例8 一款抓图软件	327
案例9 Word艺术字	333
案例10 图像特技	335
本章小结	339
 第11章 Visual Basic综合案例	 340
案例1 文件管理器	340
案例2 游戏特效——俄罗斯方块	349
案例3 酒店管理系统	367
本章小结	396



第 1 章

Visual Basic与数据库

本章内容

本章重点讲解了Visual Basic强大的数据库操作功能,利用Visual Basic可以动态地创建数据库、数据库表,可以查询数据库中的信息,并可以利用图表来表示数据信息,具体内容如下:

- ▣ 显示所有数据库表及表中的数据
- ▣ 图表显示数据库中的数据
- ▣ 动态查询数据库中的数据
- ▣ 利用DataGrid控件显示数据库中数据信息
- ▣ 利用Printer对象直接打印数据库中的数据
- ▣ 调用Excel软件实现打印预览功能
- ▣ 动态改变数据库中数据图表
- ▣ 带有图像信息的数据库数据的操作
- ▣ 带有多媒体信息的数据库数据的操作
- ▣ 动态创建访问数据库的DSN
- ▣ 动态创建数据库及数据库表
- ▣ 分页显示数据库表中的记录
- ▣ 动态显示数据库中图片信息
- ▣ 编程实现利用向导把文本文件导入到数据库中
- ▣ 本章小结



案例1 显示所有数据库表及表中的数据

案例运行效果与操作

本案例是利用Data控件连接数据库,在列表框中显示所有的数据库表,再利用MSFlexGrid控件来显示当前选择的数据库表中的所有数据,如图1-1所示。

制作要点

1. Data控件的应用
2. 列表框控件的应用
3. ActiveX控件的添加
4. MSFlexGrid控件的应用
5. SQL查询语句的应用
6. For Each循环语句的应用

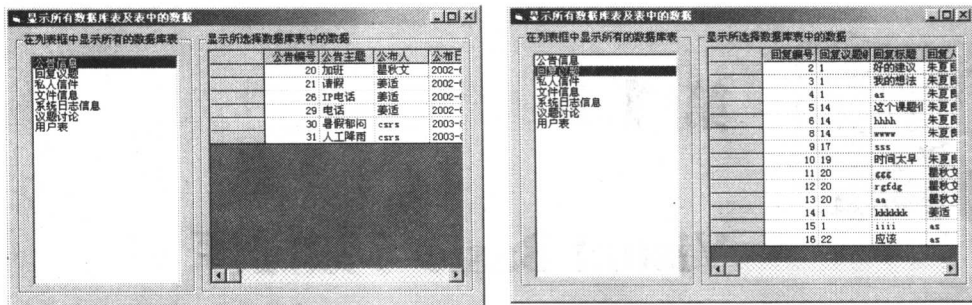


图1-1 显示所有数据库表及表中的数据

步骤详解

1. 单击菜单栏中的“文件/新建工程”命令，这时会弹出“新建工程”对话框，如图1-2所示，选择“标准EXE”项，然后单击 **确定** 按钮，这样就新建一个工程。

2. 选择窗体，设置窗体的几个属性：名称为“Form1”，Caption为“显示所有数据库表及表中的数据”，BorderStyle为“2-Sizable”，如图1-3所示。

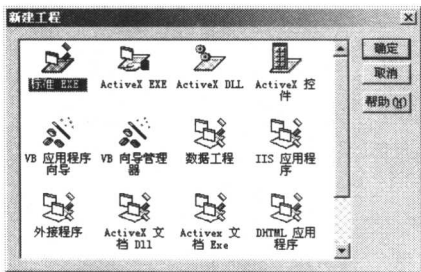


图1-2 “新建工程”对话框



图1-3 窗体属性设置

3. 单击工具箱中的 工具按钮，在窗体中按下鼠标左键进行拖动，并设置框架的Caption属性为“显示所有数据库表及表中的数据”，名称属性为“Frame1”。

4. 单击工具箱中的 工具按钮，在Frame1框架中，按下鼠标左键进行拖动，并设置其List属性为空。

5. 单击工具箱中的 工具按钮，在窗体中按下鼠标左键进行拖动，并设置框架的Visible属性为“False”，名称属性为“Data1”。

6. 添加ActiveX控件。操作方法是：在工具箱上单击右键，在弹出的菜单中单击“部件”命令，立即弹出“部件”对话框，选择“Microsoft FlexGrid Control 6.0”前面的复选框，如图1-4所示。

7. 选择好要添加的ActiveX控件后，单击“确定”按钮，就会在工具箱中看到该ActiveX工具按钮。

8. 单击工具箱中的 工具按钮，在窗体中按下鼠标左键进行拖动，并设置框架的Caption

属性为“显示所选数据库表中的数据”，名称属性为“Frame2”。

9. 单击该ActiveX工具按钮，在Frame2框架中，按下鼠标左键进行拖动，并设置其DataSource属性为“Data1”，从而实现与数据库的关联。

10. 调整各个控件实例在窗体上的位置及大小，界面最终设计效果如图1-5所示。

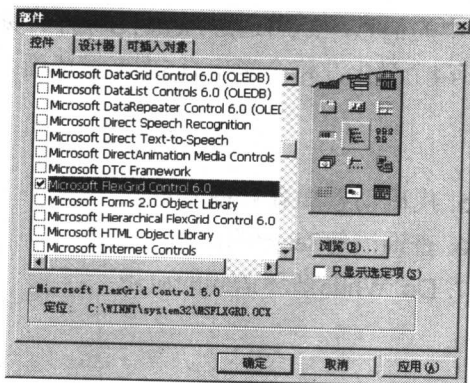


图1-4 “部件”对话框

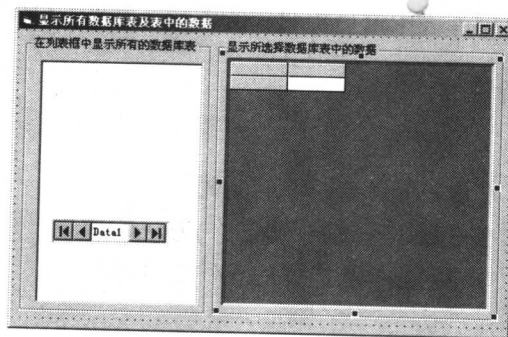


图1-5 窗体界面设计

程序源代码与解释

```
Option Explicit

Private Sub Form_Load( )
    Dim dbname As String
    Dim db As Database
    Dim qdef As QueryDef
    Dim td As TableDef
    dbname = App.Path
    If Right$(dbname, 1) <> "\" Then dbname = dbname & "\"
    dbname = dbname & "office.mdb"
    Set db = OpenDatabase(dbname)
    For Each td In db.TableDefs
        If Left$(td.Name, 4) <> "MSys" Then
            List1.AddItem td.Name
        End If
    Next td
    db.Close
    Data1.DatabaseName = dbname
End Sub

Private Sub List1_Click( )
    Dim table_name As String
    Dim sql As String
    table_name = List1.List(List1.ListIndex)
    sql = "SELECT * FROM " & table_name
    Data1.Caption = table_name
    Data1.RecordSource = sql
    Data1.Visible = False
    Data1.Refresh
    MSFlexGrid1.Visible = True
End Sub
```

//窗体加载事件

//定义字符串及数据库变量

//为变量dbname赋予当前程序目录

//所使用的数据库文件

//打开该数据库

//利用For Each循环语句向列表框中添加所有数据库表

//关闭数据库

//定义字符串变量

//把列表框中所选择的数据库表名赋给变量table_name

//SQL语句，查询当前表中的所有数据

//设置Data控件的属性

//Data控件刷新



案例2 图表显示数据库中的数据

案例运行效果与操作

本案例是利用自定义函数，动态加载数据库表中的信息，然后通过PictureBox控件来绘制数据库表中信息相对应的图形，如图1-6所示（共有4个学生，每个学生有5次考试成绩）。

制作要点

1. 自定义数据类型
2. PictureBox控件的应用
3. PictureBox控件Line()方法的应用
4. PictureBox控件Print()方法的应用
5. 代码动态加载数据库
6. 查询语句Select的应用
7. Do While循环语句的应用

步骤详解

1. 单击菜单栏中的“文件/新建工程”命令，这时会弹出“新建工程”对话框，如图1-7所示，选择“标准EXE”项，然后单击 **确定** 按钮，这样就新建一个工程。

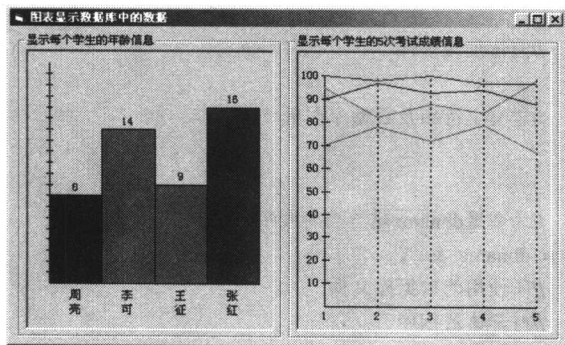


图1-6 图表显示数据库中的数据

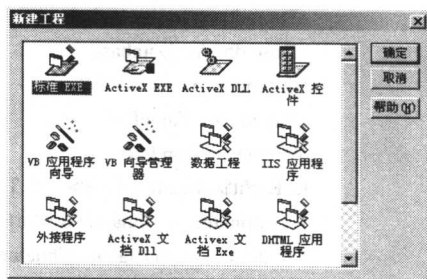


图1-7 “新建工程”对话框

2. 选择窗体，设置窗体的几个属性：名称为“Form1”，Caption为“图表显示数据库中的数据”，BorderStyle为“2-Sizeable”。

3. 单击工具箱中的 工具按钮，在窗体中按下鼠标左键进行拖动，并设置框架的Caption属性为“显示每个学生的年龄信息”，名称属性为“Frame1”。

4. 单击工具箱中的 工具按钮，在Frame1框架中按下鼠标左键进行拖动，然后设置其名称为“picAge”，再设置AutoRedraw属性为“True”。

5. 单击工具箱中的 工具按钮，在窗体中按下鼠标左键进行拖动，并设置框架的Caption属性为“显示每个学生的5次考试成绩信息”，名称属性为“Frame2”。

6. 单击工具箱中的 工具按钮，在Frame2框架中按下鼠标左键进行拖动，然后设置其名称为“picScore”，再设置AutoRedraw属性为“True”。

7. 单击工具箱中的 工具按钮，在窗体中按下鼠标左键进行拖动，并设置框架的Visible

属性为“False”，名称属性为“Data1”。

8. 调整各个控件实例在窗体上的位置及大小，界面最终设计效果如图1-8所示。

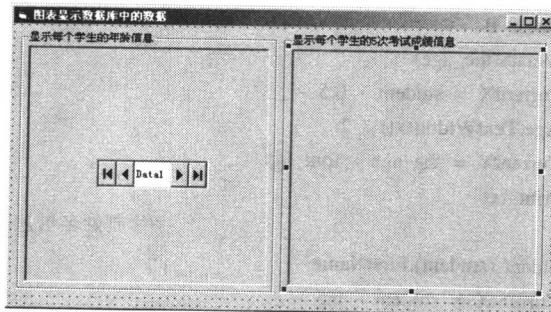


图1-8 窗体界面设计

程序源代码与解释

```
Option Explicit
Private Const NUM_TESTS = 5           //定义常量
Private Type StudentRecord            //自定义学生信息数据类型
    LastName As String
    FirstName As String
    Age As Integer
    StudentID As Integer
    Score(1 To NUM_TESTS) As Integer
End Type
Private NumStudents As Integer
Private Students( ) As StudentRecord //定义学生信息变量数组

Private Sub GraphAges( )              //自定义绘制学生年龄与姓名过程
    Dim student As Integer
    Dim lft As Single
    Dim i As Integer
    Dim the_age As Integer            //定义各种变量
    Dim txt As String
    Dim font_hgt As Single

    lft = -NumStudents / 10
    picAge.Scale (lft, 21)-(NumStudents + 0.25, -4)
    font_hgt = 1.2 * picAge.TextHeight("X")
    picAge.Line (0, 20)-(0, 0)       //绘制坐标线
    picAge.Line -(NumStudents, 0)
    For i = 1 To 19
        picAge.Line (-lft / 6, i)-Step(lft / 3, 0)
    Next i

    //绘制年龄信息图形

    For student = 1 To NumStudents
        the_age = Students(student).Age
        FillColor = QBColor(student Mod 16)
        picAge.Line (student - 1, 0)- _
            (student, the_age), _
```

```

        QBColor(student Mod 8), BF
    picAge.Line (student - 1, 0)- _
        (student, the_age), _
        vbBlack, B
    txt = Format$(the_age)
    picAge.CurrentX = student - 0.5 - _
        picAge.TextWidth(txt) / 2
    picAge.CurrentY = the_age - font_hgt
    picAge.Print txt

                                                //绘制姓名信息图形

    txt = Students(student).FirstName
    picAge.CurrentX = student - 0.5 - _
        picAge.TextWidth(txt) / 2
    picAge.CurrentY = font_hgt * 0.5
    picAge.Print txt
    txt = Students(student).LastName
    picAge.CurrentX = student - 0.5 - _
        picAge.TextWidth(txt) / 2
    picAge.CurrentY = font_hgt * 1.5
    picAge.Print txt
Next student
End Sub

Private Sub GraphScores( )
                                                //自定义绘制学生成绩过程
Dim test As Integer
Dim lft As Single
Dim i As Integer
Dim the_score As Integer
                                                //定义各种变量
Dim txt As String
Dim font_hgt As Single
Dim student As Integer
    lft = -NUM_TESTS / 10
    picScore.Scale (1 + lft, 110)-(NUM_TESTS + 0.25, -10)
    font_hgt = picScore.TextHeight("X")

                                                //绘制垂直坐标轴

    picScore.Line (1, 100)-(1, 0)
    picScore.Line -(NUM_TESTS, 0)
    For i = 10 To 100 Step 10
                                                //绘制垂直坐标刻度线
        picScore.Line (1 - lft / 6, i)-Step(lft / 3, 0)
        txt = Format$(i)
        picScore.CurrentX = 1 + lft / 4 - picScore.TextWidth(txt)
        picScore.CurrentY = i - font_hgt / 2
        picScore.Print txt
    Next i
    font_hgt = font_hgt * 1.2
    picScore.DrawStyle = vbDot
    For test = 1 To NUM_TESTS
                                                //绘制水平坐标线
        picScore.Line (test, 0)-(test, 100)
        txt = Format$(test)
        picScore.CurrentX = test - picScore.TextWidth(txt) / 2

```

```

picScore.CurrentY = font_hgt / 3
picScore.Print txt
Next test
picScore.DrawStyle = vbSolid                                //绘制学生成绩

For student = 1 To NumStudents
    picScore.ForeColor = QBColor(student Mod 8)
    the_score = Students(student).Score(1)
    picScore.CurrentX = 1
    picScore.CurrentY = the_score
    For test = 2 To NUM_TESTS
        the_score = Students(student).Score(test)
        picScore.Line -(test, the_score)
    Next test
Next student
End Sub

Private Sub LoadData( )                                     //自定义加载数据库过程
Dim ws As Workspace
Dim db As Database
Dim student_rs As Recordset
Dim score_rs As Recordset                                  //定义调用数据库的各种变量
Dim query As String
Set ws = DBEngine.Workspaces(0)
Set db = ws.OpenDatabase(App.Path & "\graphdb.mdb")        //打开数据库
                                                    //Select查询语句
query = _
"SELECT FirstName, LastName, " & _
"StudentID, Age FROM Students " & _
"ORDER BY StudentID"
Set student_rs = _
db.OpenRecordset(query, dbOpenSnapshot)
With student_rs
Do While Not .EOF                                           //利用Do While循环语句提示数据库中的姓名数据信息
    NumStudents = NumStudents + 1
    ReDim Preserve Students(1 To NumStudents)
    Students(NumStudents).LastName = !LastName
    Students(NumStudents).FirstName = !FirstName
    Students(NumStudents).Age = !Age
    Students(NumStudents).StudentID = !StudentID

    #If DEBUG_MSGS = 1 Then
        Debug.Print _
            Students(NumStudents).FirstName & " " & _
            Students(NumStudents).LastName & ", Age: " & _
            Students(NumStudents).Age & ", ID: " & _
            Students(NumStudents).StudentID
    #End If

    query = _

```

```

"SELECT TestNumber, Score FROM " & _
"TestScores WHERE StudentID=" & _
!StudentID & _
" ORDER BY TestNumber"
Set score_rs = _
db.OpenRecordset(query, dbOpenSnapshot)
Do While Not score_rs.EOF           //提取成绩数据信息
    Students(NumStudents).Score( _
        score_rs!TestNumber) = _
        score_rs!Score

    #If DEBUG_MSGS = 1 Then
        Debug.Print _
            "    Score(" & _
            Format$(score_rs!TestNumber) & ") = " & _
            Students(NumStudents).Score( _
                score_rs!TestNumber)
    #End If

    score_rs.MoveNext                //移动到下一条记录
Loop
score_rs.Close
.MoveNext

Loop
End With
student_rs.Close                    //关闭数据库的各种对象
ws.Close
End Sub

Private Sub Form_Load( )            //窗体的加载事件
    LoadData
    GraphAges
    GraphScores
End Sub

```



案例3 动态查询数据库中的数据

案例运行效果与操作

本案例利用SQL语句实现动态查询数据库中信息的功能，程序运行后，如图1-9所示。

单击下拉列表框按钮，可以选择动态查询的字段，默认为“公布人”，然后在“输入查询内容”文本框中输入查询信息，再单击“动态查询”按钮即可，如图1-10所示。

单击“显示全部”按钮，就可以全部显示数据库表中的数据。

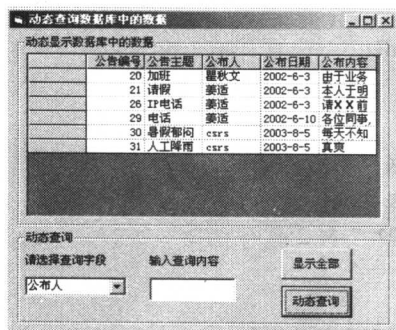


图1-9 程序运行效果

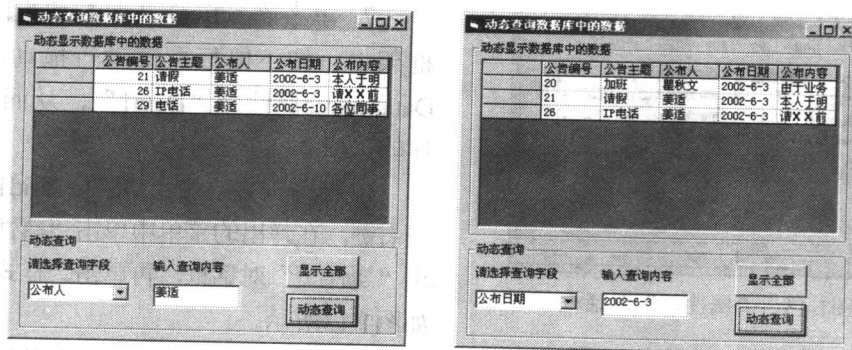


图1-10 按不同字段动态查询的效果

制作要点

1. Data控件的应用
2. 下拉列表框控件的应用
3. 数据库的动态加载
4. 带有变量的SQL语句
5. MSFlexGrid1控件的应用
6. IF条件语句的应用

步骤详解

1. 单击菜单栏中的“文件/新建工程”命令，这时会弹出“新建工程”对话框，如图1-11所示，选择“标准EXE”项，然后单击 **确定** 按钮，这样就新建一个工程。
2. 选择窗体，设置窗体的几个属性：名称为“Form1”，Caption为“动态查询数据库中的数据”，BorderStyle为“2-Sizable”。
3. 单击工具箱中的 工具按钮，在窗体中按下鼠标左键进行拖动，并设置框架的Visible属性为“False”，名称属性为“Data1”。
4. 单击工具箱中的 工具按钮，在窗体中按下鼠标左键进行拖动，并设置框架的Caption属性为“动态显示数据库中的数据”，名称属性为“Frame1”。
5. 添加ActiveX控件。操作方法是：在工具箱上单击右键，在弹出的菜单中单击“部件”命令，立即弹出“部件”对话框，选择“Microsoft FlexGrid Control 6.0”前面的复选框，如图1-12所示。
6. 选择好要添加的ActiveX控件后，单击“确定”按钮，就会在工具箱中看到该ActiveX工具按钮。

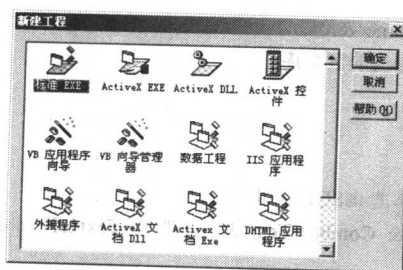


图1-11 “新建工程”对话框

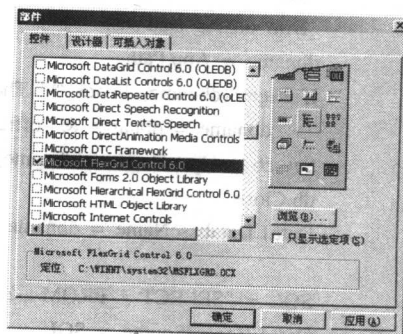


图1-12 “部件”对话框