Delphi 数据库开发
关键技术与实例应用

明日科技　赛奎春　陈紫鸿　宋　坤　编著

人 民 邮 电 出 版 社

图书在版编目(CIP)数据

需要全本请访问 www.ertongbook.com

Delphi 数据库开发关键技术与实例应用 / 明日科技，赛奎春，陈紫鸿，宋坤编著.
—北京：人民邮电出版社，2004.5
ISBN 7-115-12096-X

Ⅰ. D… Ⅱ. ①赛…②陈…③宋… Ⅲ. 软件工具—程序设计 Ⅳ. TP311.56

中国版本图书馆 CIP 数据核字（2004）第 025850 号

内 容 提 要

本书从 Delphi 用户经常遇到的问题入手，结合应用实例的讲解，帮助读者快速掌握使用 Delphi 开发数据库管理系统的关键技术、方法和技巧，从而开发出适合企业应用的数据库系统。

全书共分为 13 章，内容包括如何使用代码设计 SQL Server 数据库、如何连接数据库、数据集与数据访问组件应用技巧、数据输入输出技术、常用控制组件应用技术、SQL 数据查询技术、设计报表和图表、软件与数据库安全、计算机外围硬件程序设计、数据库典型程序设计、多层结构程序开发设计、SQL Server 2000 开发设计、使用 InstallShield 制作安装程序。

本书配套光盘中提供了书中应用实例的源程序代码，所有源程序均在 Windows 98 或 Windows XP 下调试通过，确保能够正常运行。本书内容精练、重点突出，适合 Delphi 程序开发人员阅读和参考，也可供大中专院校计算机相关专业的师生学习。

Delphi 数据库开发关键技术与实例应用

◆ 编　　著　明日科技　赛奎春　陈紫鸿　宋　坤
责任编辑　马　嘉

◆ 人民邮电出版社出版发行　　北京市崇文区夕照寺街 14 号

邮编　100061　　电子函件　315@pptph.com.cn
网址　http://www.pptph.com.cn
读者热线　010-67180876

北京汉魂图文设计有限公司制作

北京印刷厂印刷

新华书店总店北京发行所经销

◆ 开本：787×1092　1/16

印张：26.25

字数：638 千字　　2004 年 4 月第 1 版
印数：1- 000 册　　2004 年 4 月北京第 1 次印刷

ISBN 7-115-12096-3/TP・3857

定价：　.00 元（附光盘）

本书如有印装质量问题，请与本社联系　电话：（010）67129223

前　言

数据库开发是一项复杂的工作，即使开发人员熟练掌握了大量编程知识，在编写数据库管理系统时，仍然可能无从入手或无法开发出令企业满意的数据库产品。本书作者根据多年为各种类型的企事业单位开发相关数据库系统的编程经验,提炼出了使用 Delphi 开发数据库管理系统必须掌握的关键技术和技巧，通过详细的讲解和应用实例分析，帮助读者快速掌握 Delphi 数据库开发的关键技术，开发出适合企业应用的数据库系统。

为便于读者学习，本书每一章节的内容都分为“ 应用难点 ”、“ 解决思路 ”、“ 关键技术 ” 和“ 实例应用 ” 4 个部分。“ 应用难点 ” 提出了数据库开发中遇到的重点和难点问题 ;“ 解决思路 ” 给出了问题的解决思路与方法 ;“ 关键技术 ” 主要介绍解决问题的关键技术 ;“ 实例应用 ” 分析一个针对性的应用实例，读者可以结合实例上机实践。

本书配套光盘中提供了书中应用实例的源程序代码，所有源程序均在 Windows 98 或 Windows XP 下调试通过，确保能够正常运行。

本书内容

全书分为 13 章，各章主要内容如下。

第 1 章“ 使用代码设计 SQL Server 数据库 ”。以关系型数据库 SQL Server 为例，介绍通过 Delphi 编程创建数据库、数据库对象和维护数据库的方法和技巧。

第 2 章“ 如何连接数据库 ”。介绍使用 BDE 和 ADO 对象访问 SQL Server、Visual FoxPro、Access、Paradox 等数据库的方法和技巧。

第 3 章“ 数据集与数据访问组件应用技巧 ”。从使用数据集与数据访问组件遇到的重点和难点问题入手，通过 10 个典型问题的分析和解决，透彻地分析了应用数据集与数据访问组件的相关技巧。

第 4 章“ 数据输入输出技术 ”。以作者在开发管理系统时遇到的各种问题为主线，通过对数据输入输出技术的深入剖析，为读者提供问题的解决思路和方案。

第 5 章“ 常用控制组件应用技术 ”。重点介绍常用控件的关键技术与技巧。

第 6 章“ SQL 数据查询技术 ”。针对使用 Delphi 7 开发 SQL Server、Access、Paradox 数据库的开发人员，基于 SQL 语句在商业开发中的实际应用，着重解决开发人员使用 SQL 过程中遇到的关键问题。

第 7 章“ 设计报表和图表 ”。通过 TQuickReport 组件来设计多种格式的报表。这些报表格式是比较典型的报表形式，在实际程序开发中具有重要意义和作用。

第 8 章“ 软件与数据库安全技术 ”。 通过几个软件与数据库保护的实例，介绍实现软件与数据库安全的思路与方法。

第 9 章“ 计算机外围硬件程序设计 ”。从作者使用相关硬件的经验入手，结合相关硬件在商业管理系统的应用实例，介绍硬件程序设计的方法与技巧。这些硬件包括加密狗、扫描器、价格显示器、钱箱、刷卡机、扫描仪等。

第 10 章“ 数据库典型程序设计 ”。通过对商业管理系统各功能模块的分析、提炼，对一

些重要的、具有共性的典型窗体进行分析和讲解，使读者迅速掌握通用商业管理系统的开发方法。

第 11 章“ 多层结构程序开发设计 ”。以开发人员在多层结构系统开发中遇到的重点和难点问题为主线，通过对典型问题的分析与解决，介绍了多层结构程序开发的技术与技巧。

第 12 章“ SQL Server 2000 开发设计 ”。从 Delphi 开发人员的角度，通过 8 个完整实例，介绍使用 SQL Server 2000 数据库作为后台数据库时应掌握的技术，包括创建数据库和表、生成 SQL 脚本、创建存储过程、创建触发器、导入导出数据、数据备份和数据恢复等。

第 13 章“ 使用 InstallShield 制作安装程序 ”。对使用 Install Shield 打包发行软件的完整过程进行介绍，包括组织安装信息、配置程序数据、设置目标系统、设置安装画面、设置系统需求、安装与部署等内容。

本书特点

✓ 突出关键，注重实效。本书从开发数据库系统过程中容易遇到的技术难点入手，总结出开发数据库系统应该掌握的关键技术和技巧。通过对这些关键技术的讲解和应用实例的分析，帮助读者有效地解决工作中的问题。

✓ 构思独特，结构严谨。本书按照技术问题的分类来谋篇布局，通过对某一类问题的深入分析，使读者在掌握关键技术的基础上，达到触类旁通，举一反三的效果。

✓ 思路清晰，分析透彻。全书紧密围绕具体的开发任务，不仅注重解决具体的编程疑难问题，还能够帮助读者形成发现问题、分析问题、解决问题的编程习惯。

✓ 案例丰富，步骤详细。针对每一个技术难点都给出了相应的应用实例。实例全部来自于作者的工程实践，突出了商业实用性和针对性，实例讲解力求步骤详尽，清晰流畅。

技术支持

本书由明日科技公司策划和组织编写，赛奎春、陈紫鸿、宋坤编写。特别感谢邹天思、刘文娟、于海娇、王国辉、阎小泽、邹淑芳等几位同事在本书编写过程中承担了大量的资料整理和排版校对工作。

编写本书的想法来自于作者多年从事企业级数据库系统开发工作的经验总结和案例积累，由于作者的水平所限，书中疏漏之处在所难免，敬请广大读者批评指正。如果您有任何的问题或建议，请登录网站 :http://www.mingrisoft.com，我们将为您提供周到的技术支持。

作　者

目　录

第 1 章　使用代码设计 SQL Server 数据库 ····· 1

1.1　如何使用代码创建数据库 ····· 1
1.2　如何使用代码定义表和字段 ····· 4
1.3　如何使用代码定义索引 ····· 6
1.4　如何使用代码创建存储过程 ····· 8
1.5　如何使用代码创建触发器 ····· 11
1.6　如何使用代码压缩数据库 ····· 14
1.7　如何使用代码分离数据库 ····· 16
1.8　如何使用代码附加数据库 ····· 17

第 2 章　如何连接数据库 ····· 20

2.1　如何使用 BDE 访问 Paradox 数据库 ····· 20
2.2　如何使用 BDE 访问 Access 和 FoxPro 数据库 ····· 25
2.3　如何使用 BDE 访问 SQL Server 数据库 ····· 29
2.4　如何动态创建 ODBC 数据源 ····· 32
2.5　如何使用 ODBC 连接 SQL Server 数据库 ····· 35
2.6　如何使用 ADO 访问 Paradox 数据库 ····· 39
2.7　如何使用 ADO 访问 Access 数据库 ····· 42
2.8　如何使用 ADO 访问带有密码的 Access 数据库 ····· 45

第 3 章　数据集与数据访问组件应用技巧 ····· 49

3.1　利用 TDatabase 组件去掉注册对话框 ····· 49
3.2　对数据表的字段进行计算 ····· 52
3.3　使用 TTable 组件的 Filter 方法动态过滤数据 ····· 55
3.4　利用 Locate 进行多个字段动态搜索 ····· 57
3.5　如何在 TQuery 组件中赋值和运行 SQL ····· 60
3.6　如何在 TQuery 中赋值变量 ····· 63
3.7　在 TQuery 组件中组装 SQL 实现动态查询 ····· 65
3.8　如何利用 TADOTable 组件设计主从明细表 ····· 68
3.9　如何使用 TADOQuery 组件设计主从明细表 ····· 70
3.10　如何利用 TADODataSet 对象批量更新数据 ····· 73

第 4 章　数据库输入输出技术 ····· 78

需要全本请访问 www.ertongbook.com

4.1 使用数据控制组件实现数据记录的添加和修改 …… 78
4.2 使用非数据组件实现数据记录的添加和修改 …… 81
4.3 在数据输入时验证数据是否合法 …… 85
4.4 在数据输入时检测输入是否重复并提示 …… 89
4.5 对组件内数据进行数据格式化处理 …… 92
4.6 利用日期组件实现日期录入 …… 96
4.7 在数据输入时自动切换输入法 …… 98
4.8 利用键盘 Enter、↑、↓实现组件间的灵活跳转 …… 101
4.9 如何自动生成产品编号 …… 105
4.10 利用组件数组实现数据的录入 …… 108
4.11 如何保存图片数据到数据库 …… 111
4.12 如何获得汉字的拼音简码 …… 113

第 5 章 常用控制组件应用技术 …… 117

5.1 如何将字段内容添加到 TComboBox 组件中 …… 117
5.2 利用 TDBGrid 组件实现字段间计算 …… 120
5.3 在 TDBGrid 单元格中显示图片数据 …… 123
5.4 在 TDBGrid 组件中显示多数据表数据 …… 125
5.5 在程序运行时设置 TStringGrid 组件行列 …… 128
5.6 在程序中设置 TStringGrid 组件颜色 …… 131
5.7 使用 TTreeView 实现数据库的树状显示 …… 134
5.8 使用 TListView 实现数据库的图形显示 …… 137
5.9 在程序运行时对 TStringGrid 单元格数据进行计算 …… 140
5.10 利用 TStringGrid 实现表单式输入数据 …… 144
5.11 使用 TDBCtrlGrid 组件实现分栏显示数据 …… 150
5.12 利用决策组组件统计分析数据 …… 152

第 6 章 SQL 数据查询技术 …… 157

6.1 利用 WHERE 参数过滤数据 …… 157
6.2 使用通配符进行模糊查询 …… 160
6.3 利用关键字 DISTINCT 去除重复记录 …… 163
6.4 使用 AND、OR 和 NOT 进行复合条件查询 …… 166
6.5 在查询程序中使用常量 …… 168
6.6 在查询程序中使用组件 …… 171
6.7 如何对某一区间数据进行查询 …… 174
6.8 如何对日期数据进行查询 …… 176
6.9 利用聚集函数对数据进行汇总 …… 179
6.10 如何将查询结果分组小计 …… 182
6.11 如何对查询结果进行排序 …… 185

6.12　追加查询结果到已存在表 ····· 187
6.13　对数据库数据进行局部更新 ····· 190
6.14　对数据库数据进行局部删除 ····· 192
6.15　如何将查询结果生成新表 ····· 194

第 7 章　设计报表和图表 ····· 197

7.1　利用报表生成器生成报表 ····· 197
7.2　设计标签式报表 ····· 201
7.3　设计汇款单式报表 ····· 206
7.4　设计分组式报表 ····· 209
7.5　设计主明细式报表 ····· 213
7.6　设计图案式报表 ····· 217
7.7　利用 Excel 打印报表 ····· 220
7.8　设计与修改图表 ····· 227

第 8 章　软件与数据库安全技术 ····· 232

8.1　如何设置对 Access 数据库的访问权限 ····· 232
8.2　如何利用 INI 文件对软件进行注册 ····· 235
8.3　如何利用注册表设计软件注册程序 ····· 237
8.4　如何跟踪操作员操作 ····· 240
8.5　为不同用户设置读写权限 ····· 247
8.6　如何设计软件试用程序 ····· 251
8.7　利用硬件生成产品序列号 ····· 257

第 9 章　计算机外围硬件程序设计 ····· 260

9.1　如何将密码写入加密狗 ····· 260
9.2　如何使用加密狗进行程序加密 ····· 264
9.3　利用扫描器入库和销售商品 ····· 269
9.4　如何设计客户价格显示器程序 ····· 278
9.5　如何显示和设计条形码程序 ····· 283
9.6　如何设计钱箱控制程序 ····· 287
9.7　如何设计两台计算机串口通信程序 ····· 290
9.8　如何设计扫描仪控制程序 ····· 293
9.9　如何设计刷卡机程序 ····· 294

第 10 章　数据库典型程序设计 ····· 298

10.1　如何设计主程序窗体 ····· 298
10.2　文本式录入程序设计 ····· 303
10.3　表单式录入程序设计 ····· 307

10.4 通用修改程序设计 · · ·

第 1 章 使用代码设计 SQL Server 数据库

数据库技术产生于 20 世纪 60 年代中期，数据库（DataBase）简称 DB，是长期存储在计算机内有组织的、可共享数据的集合。目前市场上所用的数据库多数都是关系型数据库，例如 Access、Paradox、DB2、Oracle、SQL Server 等。关系型数据库是以关系模型为基础的数据库，它是应用数学理论处理数据的一种组织方法，是数据库设计中的一种新思想。本章将以关系型数据库 SQL Server 为例介绍使用 Delphi 编程创建数据库、数据库对象以及维护数据库的一些方法、原则和技巧。创建数据库、数据库对象包括创建数据库、数据表、字段、索引、存储过程和触发器；维护数据库包括压缩、分离、附加数据库。

1.1 如何使用代码创建数据库

1 应用难点

在使用 SQL Server 创建数据库时，通常是在企业管理器或查询分析器中创建完成的。是否可以利用 Delphi 编程创建数据库呢？

2 解决思路

在 Delphi 程序中使用 SQL 语句可以创建数据库。SQL（Structured Query Language）是一种结构化查询语言，是组织、管理和检索计算机中数据库数据的工具。本节将对 SQL 语句中的“Create DataBase”语句进行讲解，有关 SQL 语句的知识在后面章节中会做详细的介绍。

注意：如果创建与已经存在的数据库同名的数据库，系统会报错。

3 关键技术

要在 Delphi 中使用 SQL 语句，可以使用数据集组件 TQuery，该组件在 BDE 组件面板中。TQuery 组件有一个 SQL 属性，在程序中要使用 SQL 语句，就需要将 SQL 语句添加到该属性中。将 SQL 语句添加到该属性中有两种方法。

（1）在程序设计时，用鼠标单击【TQuery】组件，在对象观察器中将显示 TQuery 组件的属性和事件，选中 SQL 属性，双击右边的单元格，将显示如图 1.1 所示对话框。

可以将 SQL 语句写入该编辑框中。

（2）在程序中将 SQL 语句添加到 TQuery 组件的 SQL 属性中，代码如下：

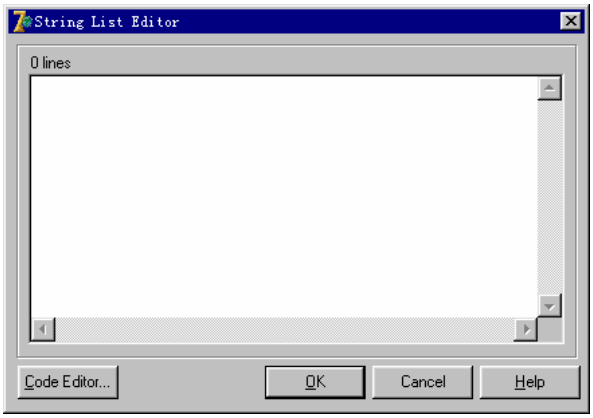

图 1.1　字符串编辑框

```
with Query1 do
begin
  Close;      //关闭与数据库连接
  SQL.Clear;      //清空 SQL 语句
  SQL.add(' select * from  学生表');     //将 SQL 语句“ select * from 学生表 ”添加到 SQL 中
  Open;       //打开与数据库的连接，执行 SQL 语句
end;
```

本节要通过编程创建数据库，需要使用 SQL 语句中的 **Create DataBase** 语句，下面对该语句进行详细介绍。

Create DataBase 语法格式为：

```
Create DataBase databasename
On
NAME = logical_file_name ,
FILENAME = os_file_name
```

其中 databasename 指定要创建数据库的名称，数据库名称在服务器中必须是惟一的。logical_file_name 表示用来在创建数据库后执行的 Transact-SQL 语句中引用文件的名称。logical_file_name 在数据库中必须惟一，并且符合标识符的规则。os_file_name 表示操作系统创建<filespec>定义的物理文件时使用的路径名和文件名，它不能指定压缩文件系统中的目录。

注意：要使用 Delphi 编程创建数据库，要注意以下几点：

① 使用 TDatabase 组件连接数据，用以保存登录数据库所需的身份认证数据，包括用户名和密码。

② 使用 Query1 组件创建数据库，要设置 Query1 组件的 Databasename 与 TDatabase 组件的 Databasename 一致。因为 TDatabase 组件已保存了认证数据，执行时可免去 Query1 组件的认证操作。

4　实例应用

本节实例主要实现在程序中创建 SQL Server 数据库。运行程序，单击【创建】按钮，将创建一个名称为“明日科技”的关系数据库。效果如图 1.2 所示。

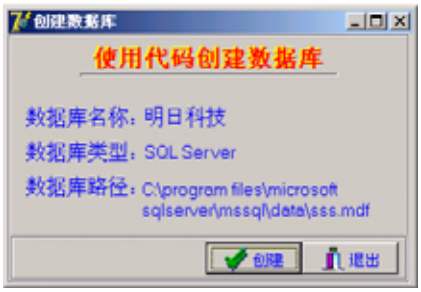

图 1.2　创建数据库

具体步骤如下：

（1）启动 Delphi，选择【File】菜单中的【New Application】新建项目，选择【File】菜单中的【Save Project as】菜单项，将工程保存为 gc1.dpr。

（2）在新建的窗体上放置 TBitBtn、TLabel、TQuery、TBevel、TPanel 等组件。设置各主要组件对象的属性，如表 1.1 所示。

表 1.1　　主要组件对象的属性列表

对　象　名	对象类型	属　　性	值
Database	Tdatabase	Aliasname	Mqis
		Databasename	Mq
		Corrected	True
		Keepcorrection	False
		Loginprompt	False
		Params	User name=sa　Password=
Form1	Tform	Caption	创建数据库
Query1	Tquery	DataBaseName	CYJL

（3）添加如下程序代码。

```
    procedure TForm1.BitBtn1Click(Sender: TObject);
    begin
      Try
        with Query1 do
        begin
          Close;
          SQL.Clear;      //清空 SQL 内容
          SQL.Add('Create Database 明日科技 on');      //添加 SQL 内容
          SQL.Add('  (Name  =  sss  ,    Filename  =  ''c:\program  files\microsoft  sql
server\mssql\data\sss.mdf'')') ;
```


需要全本请访问 www.ertongbook.com

```
        ExecSQL;      //运行 SQL
      end;
      Application.MessageBox('创建成功。','提示',64);     //提示创建数据库成功
    Except
      Application.MessageBox('创建失败。','提示',64);     //提示创建数据库失败
      Close;
    end;
  end;
```

技巧：在设计窗体界面时，如果用鼠标来对齐组件可能不会很准确，此时可以用【Ctrl+方向键】来移动组件，该方法以 1 个像素为单位移动。第二种方法是调整窗体中网格线的间距，在窗体中，Delphi 默认的网格水平和垂直间距是 8 个像素，可以在环境选项中进行重新设置，具体操作是选择【Tools】/【Environment Options…】菜单命令，打开环境选项窗口，选择【Designer】选项卡，将 Grid size X（*x* 轴的网格间距）、Grid size Y（*y* 轴的网格间距）调整为较小值，如图 1.3 所示。

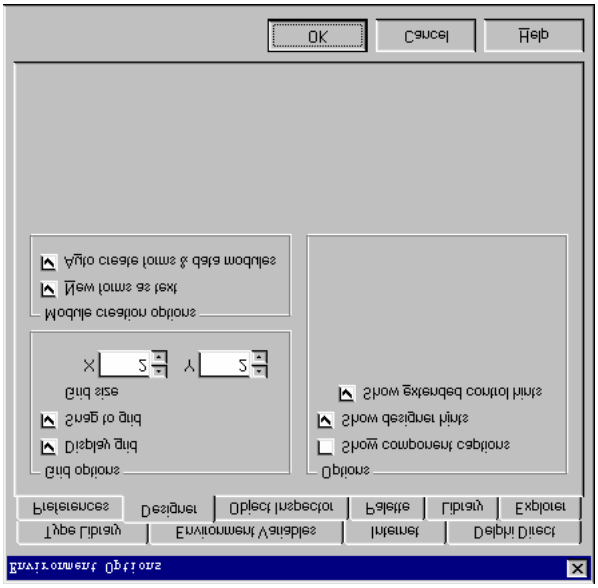

图 1.3　环境选项

1.2　如何使用代码定义表和字段

1　应用难点

在设计数据库应用程序时，经常在程序设计期间定义表结构，包括字段名称、字段类型、默认值等，那么能否在程序运行过程中动态地创建数据表和字段呢？

2 解决思路

在程序中创建数据表有两种方法，一是使用数据集组件 TTable 的 CreateTable 方法，另一种方法是通过 SQL 语句 Create Table 来创建数据表。本节采用 SQL 语句的方法来创建数据表。

3 关键技术

在程序中创建 SQL Server 数据表需要使用 Create Table 关键字。其语法格式为：

```
CREATE TABLE
  [ database_name.[ owner ] .| owner.] table_name
  {       <       column_definition       >       |       column_name       AS
computed_column_expression
  | < table_constraint >}
    [ ON { filegroup | DEFAULT } ] [ TEXTIMAGE_ON { filegroup | DEFAULT } ]
```

database_name 为创建的数据表所在的数据库名称，如果省略该参数，表明在当前数据库中创建数据表。

owner 是创建数据表的所有者。

table_name 为数据表名称，在一个数据库中，数据表名称必须惟一。

column_name 为表中的列名，在同一个表中，列名必须惟一。

computed_column_expression 是定义计算列值的表达式。计算列是物理上并不存储在表中的虚拟列。计算列由同一表中的其他列通过表达式计算得到。

ON { filegroup | DEFAULT }指定存储表的文件组，如果指定 DEFAULT，或者根本未指定 ON 参数，则表存储在默认文件组中。

TEXTIMAGE_ON 是表示 Text、NText 和 Image 列存储在指定文件组中的关键字。如果表中没有 Text、NText 或 Image 列，则不能使用 TEXTIMAGE_ON。

4 实例应用

本节实例主要实现在程序中动态创建数据表。运行程序，单击【创建】按钮，将按窗体上显示内容创建数据表。效果如图 1.4 所示。



图 1.4　创建数据表和字段

具体步骤如下：

（1）启动 Delphi，选择【File】菜单中的【New Application】新建项目，选择【File】菜单中的【Save Project as】菜单项，将工程保存为 gc2.dpr。

（2）在新建的窗体上放置 TBitBtn、TPanel、TQuery、TLabel、TShape、TBevel 等组件。设置各主要组件对象的属性，如表 1.2 所示。

表 1.2　　主要组件对象的属性列表

对 象 名	对 象 类 型	属　　性	值
Database	TDatabase	Aliasname	Mqis
		Databasename	Mq
		Corrected	True
		Keepcorrection	False
		Loginprompt	False
		Params	User name=sa　Password=
Form1	TForm	Caption	创建数据表和字段
Query1	TQuery	DataBaseName	MQIS

（3）添加如下程序代码。

```
procedure TForm1.BitBtn1Click(Sender: TObject);
begin
  Try
    with Query1 do
    begin
      Close;      //关闭 Query1
      SQL.Clear;       //清除 SQL
      SQL.Add('Create Table 学生表');    //添加 SQL
      SQL.Add('(学生编号 Varchar(20) not null,学生姓名 Varchar(20),语文成绩 int,');
      SQL.Add(' 数学成绩 int, 班主任 Varchar(20))');
      ExecSQL;      //执行 SQL
    end;
    Application.MessageBox('数据表创建成功。','提示',0+64);
  Except
    Application.MessageBox('系统出错。','提示',0+64);
  end;
end;
```

注意：在创建数据表时，字段与字段之间一定要用逗号分开，数据表中最后一个字段没有逗号，表中字段的定义必须在括号内。

1.3　如何使用代码定义索引

1　应用难点

在设计数据库应用程序中，为了提高查询速度，经常在表中建立索引。那么在程序中如何定义字段的索引呢？

2　解决思路

索引经常在创建数据表时定义，也可以在修改数据表时创建。在 1.2 节介绍 Create Table 语句中，有一个< table_constraint >选项，在该选项中便可以建立表的索引。

3　关键技术

索引是对数据库表中一个或多个列的值进行排序的结构。索引提供指针以指向存储在表中指定列的数据值，再根据指定的排序次序排列这些指针。数据库使用索引的方式与书的目录很相似：通过搜索索引找到特定的值，然后跟随指针到达包含该值的行。使用索引可以加快数据搜索的速度，但并不是数据表中每个字段都需要建立索引，因为当在插入或删除数据时，服务器必须付出一些时间来更新索引，如果表中索引过多，服务器会耗费许多资源来管理索引，最终导致系统性能降低。

创建索引有多种方法，在程序中为字段定义主键后，数据库管理系统会自动创建主键索引。此外还可以使用 Create Index 语句来建立索引。

4　实例应用

本节实例主要实现在程序中创建索引。运行程序，单击【创建】按钮，程序将会为教师表创建索引。效果如图 1.5 所示。



图 1.5　如何定义索引

具体步骤如下：

(1) 启动 Delphi，选择【File】菜单中的【New Application】新建项目，选择【File】菜单中的【Save Project as】菜单项，将工程保存为 gc3.dpr。

（2）在新建的窗体上放置 TBitBtn、TPanel、TBevel、TQuery、TLabel、TShape 等组件。设置各主要组件对象的属性，如表 1.3 所示。

表 1.3　　主要组件对象的属性列表

对　象　名	对象类型	属　　性	值
Database	TDatabase	Aliasname	Mqis
		Databasename	mq

续表

对　象　名	对象类型	属　　性	值
		Corrected	True
		Keepcorrection	False
		Loginprompt	False
		Params	User name=sa　Password=
Form1	TForm	Caption	如何定义索引
Query1	TQuery	DataBaseName	MQIS

（3）添加如下程序代码。

```
procedure TForm1.BitBtn1Click(Sender: TObject);
begin
  Try
    with Query1 do
    begin
      Close;
      SQL.Clear;
      SQL.Add('Create table 教师表');
      SQL.Add('(教师编号 Varchar(20) not null,Constraint aaaa primary key(教师编号) ,') ;
      SQL.Add('教师姓名 Varchar(20),科室 Varchar(20),年龄 int,工资  int)');
      ExecSQL;
    end;
    Application.MessageBox('索引创建成功。','提示',64);
  Except
    Application.MessageBox('系统出错。','提示',64);
  end;
end;
```

技巧：在创建数据表时，如果使用 Primary Key 为数据表建立主键索引，可以在字段定义之后添加 PrimaryKey 关键字。例如本节实例中为教师编号建立主键索引，SQL 语句可以写成“教师编号 Varchar(20) not null primary key”。

1.4　如何使用代码创建存储过程

1　应用难点

在开发数据库应用程序时，经常需要进行相同的操作，例如对数据表的查询，每次都需要写相同的 SQL 语句，程序代码就会冗余，而且执行效率不高。那么如何能精简代码并提高执行效率呢？

2　解决思路

可以将重复的 SQL 语句创建为存储过程。存储过程是一组预先被编译的 SQL 语句的集合。使用存储过程可以实现某些特定的功能，例如实现数据的插入、修改、删除、查询等。

3　关键技术

存储过程分为系统存储过程、扩展存储过程和用户自定义存储过程。

系统存储过程是数据库管理系统为用户提供的存储过程，用户不需要定义，可以直接执行系统的存储过程。在 SQL Server 中系统存储过程以 sp_开头，系统存储过程通常用来进行数据库系统的各项设置或调用系统的各项功能，在后面小节中实现数据库的附加和分离就需要调用系统存储过程。

扩展存储过程是用传统的程序设计语言编写而成的，该存储过程是以动态链接库的形式存在的。

用户自定义存储过程是用户使用 SQL 语句创建的存储过程，该存储过程会被加载到数据库的存储过程项目中。

在 SQL Server 中创建存储过程需要使用 SQL 语句 Create Procedure。下面将对 Create Procedure 语句的用法做详细的介绍。

Create Procedure 语法格式为：

```
CREATE PROC [ EDURE ] procedure_name [ ; number ]
  [ { @parameter data_type }
    [ VARYING ] [ = default ] [ OUTPUT ]
  ] [ ,...n ]
 [ FOR REPLICATION ]
AS sql_statement [ ...n ]
```

procedure_name 为创建的存储器过程名称，该名称在数据库中必须是惟一的。

; number 是可选的整数，用来对同名的过程分组，当删除存储过程时，如果不指定该选项，会将该组存储过程全部删除。

@parameter 为存储过程中的参数，在创建存储过程时也可以不指定参数。

data_type 为参数的数据类型。所有数据类型（包括 text、ntext 和 image）均可以用作存储过程的参数。

需要全本请访问 www.ertongbook.com