

第 1 章 数据库系统的基础知识

本章学习目标

本章从数据库系统的基础知识入手，对数据库的基本概念、数据模型、数据库体系结构、数据库设计及 Access 2003 数据库进行介绍。通过本章的学习，读者应该掌握以下内容：

- 数据库系统的概念与组成
- 数据模型
- 数据库体系结构
- 数据库设计方法与步骤
- Access 2003 数据库的特点

1.1 数据库系统概述

数据库技术产生于 20 世纪 60 年代末，是数据管理的最新技术，计算机科学的重要分支。在当今信息社会中，信息已成为各个行业、部门的重要财富和资源，信息系统也越来越显示出它的重要性。数据库技术是信息系统的核心和基础，它的出现极大地促进了计算机应用向各行各业的渗透。从一般的小型事务处理到大型的信息系统，越来越多的新应用领域开始采用数据库技术存储与处理其信息资源。数据库的建设规模、数据库信息量的大小和使用频度已成为衡量一个国家信息化程度的重要标志。

1.1.1 数据库系统的组成

数据(Data)是数据库中存储的基本对象。数据的种类很多，例如文字、图形、图像和声音等都是数据。

数据可定义为描述事物的符号记录。数据有多种形式，它们均可以经过数字化后储存到计算机中。在描述事物的过程中，数据与其解释是不可分的。

数据库是指长期储存在计算机内的、有组织的、可共享的数据集合。数据库中的数据是按一定的数据模型组织、描述和储存的，具有较小的冗余度、较高的数据独立性和易扩展性，并且可以被多个用户、多个应用程序共享。

数据库管理系统(Database Management System, DBMS)是位于用户与操作系统(OS)之间的一层数据管理软件，是数据库系统的中心枢纽。数据库管理系统能科学地组织和存储数据、高效地获取和维护数据。用户对数据库进行的各种操作，如数据库的建立、使用和维护，都是在 DBMS 的统一管理和控制下进行的。

数据库管理系统的主要功能有以下几个方面：

(1) 数据定义功能

提供数据定义语言(DDL), 用于定义数据库中的数据对象。

(2) 数据操纵功能

提供数据操纵语言(DML), 用于操纵数据实现对数据库的基本操作, 例如查询、插入、删除和修改等。

(3) 数据库的运行管理

保证数据的安全性、完整性、多用户对数据的并发使用及发生故障后的系统恢复。

(4) 数据库的建立和维护功能

提供数据库数据输入、批量装载、数据库转储、介质故障恢复、数据库的重组织及性能监视等功能。

数据库系统(Database System, DBS)是指在计算机系统中引入数据库之后组成的系统, 是用来组织和存取大量数据的管理系统。数据库系统是由计算机系统(硬件和基本软件)、数据库、数据库管理系统(及其开发工具)、应用系统和有关人员(数据库管理员、应用设计人员、最终用户)组成的具有高度组织性的总体。

通常情况下, 把数据库系统简称为数据库。数据库系统组件之间的关系如图 1.1 所示。

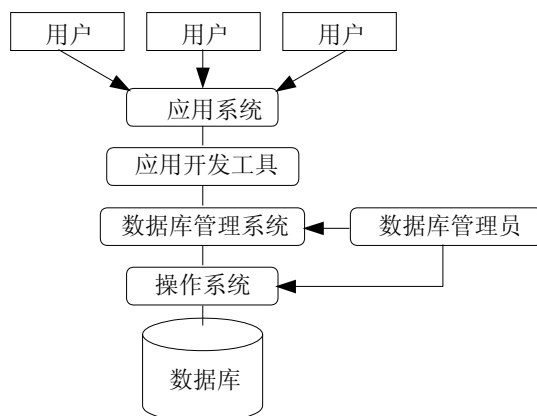


图 1.1 数据库系统组成

数据库技术的核心任务是数据处理。数据处理是指对各种数据进行收集、存储、加工和传播等一系列活动的总和。**数据管理则是指对数据进行分类、组织、编码、存储、检索和维护，它是数据处理的中心问题。**

数据管理技术的发展, 与计算机硬件(主要是外部存储器)、系统软件及计算机应用的范围有着密切的联系。数据管理技术的发展经历了以下几个阶段: 人工管理阶段、文件系统阶段、数据库系统阶段和分布式数据库系统阶段。

20 世纪 50 年代中期以前, 计算机主要用于科学计算。那时的计算机硬件方面, 外存只有卡片、纸带及磁带, 没有磁盘等可直接存取的存储设备; 软件方面, 只有汇编语言, 没有操作系统和高级语言, 更没有管理数据的软件; 数据处理的方式是批处理。这些决定了当时的数据管理只能依赖人工来进行。

人工管理阶段的特点是:

(1) 数据不进行保存。当时的计算机主要用于科学计算, 一个程序对应一个数据。在

计算某一问题时，把程序和对应的数据装入，计算完就退出，没有将数据长期保存的必要。

(2) 没有专门的数据管理软件。数据需要由应用程序自己管理，因此应用程序的设计者不仅要考虑数据的逻辑结构，还考虑数据的物理结构，比如存储结构、存取方法、输入输出方式等。存储结构发生变化，应用程序也要做相应的修改，程序员的负担非常重，数据的独立性也很差。

(3) 数据面向应用。一组数据对应一组程序。倘若多个程序使用相同的数据，必须各自定义，不能共享。所以程序之间存在大量的数据冗余。

(4) 只有程序的概念，基本上没有文件的概念。

20 世纪 60 年代，计算机技术有了很大提高，计算机的应用范围不断扩大，不仅用于科学计算，还大量用于管理。这时的计算机硬件已经有了磁盘、磁鼓等直接存取的外存设备；软件则有了操作系统、高级语言，操作系统中的文件系统专门用于数据管理；处理方式不仅有批处理，还增加了联机实时处理。

文件系统阶段的特点如下：

(1) 数据可以长期保存在磁盘上。用户可以反复对文件进行查询、修改、插入和删除等操作。

(2) 文件系统提供了数据与程序之间的存取方法。应用程序和数据有了一定的独立性，数据存储结构的改变也不一定反映在程序上，大大减轻了程序员的负担。

(3) 数据冗余量大。文件系统中，文件仍然是面向应用的，一个文件基本上对应于一个应用程序，不能对数据项进行共享，因此数据冗余大，存储空间浪费。由于数据可能有多个副本，对其中之一进行修改时还容易造成数据的不一致性。

(4) 文件之间缺乏联系，相对孤立，仍然不能反映客观世界各个事物之间错综复杂的联系。

数据库系统阶段的特点如下：

这个阶段基本实现了数据共享，减少了数据冗余，数据库采用特定的数据模型，具有较高的数据独立性，有统一的数据控制和管理功能。

分布式数据库系统阶段的特点如下：

分布式数据库在逻辑上是一个整体，是分布在不同地理位置上的数据集合，它受分布式数据库管理系统的控制和管理。

(1) 分布透明性

(2) 局部与集中控制相结合

(3) 高可靠性和可用性

(4) 高效率和灵活性

1.1.2 数据库系统的特点

20 世纪 60 年代末以来，计算机的应用更为广泛，用于数据管理的应用系统规模也更为庞大，由此带来数据量的急剧膨胀；计算机磁盘技术有了很大发展，出现了大容量的磁盘；在处理方式上，联机实时处理的要求更多。这些变化促使了数据管理手段的进步，数据库技术应运而生。与人工管理和文件系统相比，数据库系统的特点主要有以下几个方面：

(1) 数据的结构化

在文件系统中，只考虑了同一文件记录内部数据项之间的联系，而不同文件的记录之间是没有联系的，从整体上看数据是无结构的，不能反映客观世界各种事物之间的错综复杂的联系。在数据库系统中，实现了整体数据的结构化，把文件系统中简单的记录结构变成了记录和记录之间的联系所构成的结构化数据。在描述数据时，不仅要描述数据本身，还要描述数据之间的联系。

(2) 数据共享性好

数据库系统从整体角度看待与描述数据，使数据不再面向某个应用而是面向整个系统，这些数据可以供多个部门使用，实现了数据的共享。各个部门的数据基本上没有重复的存储，数据的冗余量小。

(3) 数据独立性好

数据库系统有三层结构：用户(局部)数据的逻辑结构、整体数据的逻辑结构和数据的物理结构。在这三层结构之间，数据库系统提供了两层映像功能。首先是用户数据逻辑结构和整体数据逻辑结构之间的映像，这一映像保证了数据的逻辑独立性：当数据库的整体逻辑结构发生变化时，通过修改这层映像可使局部的逻辑结构不受影响，因此不必修改应用程序。另外一级映像是整体数据逻辑结构和数据物理结构之间的映像，它保证了数据的物理独立性：当数据的存储结构发生变化时，通过修改这层映像可使数据的逻辑结构不受影响，因此应用程序同样不必修改。

(4) 数据存储粒度小

在文件系统中，数据存储的最小单位是记录；而在数据库系统中，数据存储的粒度可以小到记录中的一个数据项。因此数据库中数据存取的方式非常灵活，便于对数据的管理。

(5) 数据管理系统对数据进行统一的管理和控制

DBMS 不仅具有基本的数据管理功能，还具有如下的控制功能：

① 保证数据的完整性

数据的完整性指数据的正确性、有效性和相容性，要求数据在一定的取值范围内或相互之间满足一定的关系。例如，规定考试的成绩在 0 分到 100 分之间，血型只能是 A 型、B 型、AB 型、O 型中的一种等。

② 保证数据的安全性

让每个用户只能按指定的权限访问数据，防止不合法地使用数据，造成数据的破坏和丢失。比如学生对于课程的成绩只能进行查询，不能修改。

③ 并发控制

对多用户的并发操作加以协调和控制，防止多个进程同时存取、修改数据库中的数据时发生冲突、造成错误。比如在学生选课系统中，某门课只剩下最后一个名额，但有两个学生在两台选课终端上同时发出了选这门课的请求，必须采取某种措施，确保两名学生不能同时拥有这最后的一个名额。

④ 数据库的恢复

当数据库系统出现硬件或软件故障时，DBMS 具有把数据库恢复到最近某个时刻的正确状态上来的能力。

(6) 为用户提供了友好的接口

用户可以使用交互式的命令语言, 如 SQL(Structured Query Language, 结构化查询语言), 对数据库进行操作; 也可以把普通的高级语言(如 C++语言等)和 SQL 结合起来, 从而把对数据库的访问和对数据的处理有机地结合在一起。

1.1.3 高级数据库阶段

1. 分布式数据库系统

20 世纪 70 年代后期之前, 数据库系统多数是集中式的。分布式数据库系统是数据库技术和计算机网络技术相结合的产物, 在 80 年代中期已有商品化产品问世。分布式数据库是一个逻辑上统一、地域上分布的数据集合, 是计算机网络环境中各个结点局部数据库的逻辑集合, 同时受分布式数据库管理系统的控制和管理。

分布式数据库在逻辑上像一个集中式数据库系统, 实际上数据存储处于不同地点的计算机网络的各个结点上。每个结点都有自己的局部数据库管理系统, 使每个节点有很高的独立性。用户可以由分布式数据库管理系统(网络数据库管理系统), 通过网络通信相互传输数据。分布式数据库系统有高度透明性, 每台计算机上的用户并不需要了解他所访问的数据究竟在什么地方, 就像在使用集中式数据库一样。其主要优点有:

(1) 局部自主

网络上每个结点的数据库系统都具有独立处理本地事物的能力(大量的), 而且各局部结点之间也能够互相访问、有效地配合处理更复杂的事物。因此, 分布式数据库系统特别适合各个部门的地理位置分散的组织机构。例如, 银行业务、飞机订票和企业管理等。

(2) 可靠性和可用性

分布式系统比集中式系统有更高的可靠性, 在个别结点或个别通信链路发生故障的情况下可以继续工作。一个局部系统发生故障不至于导致整个系统停顿或破坏, 只要有一个结点上的数据备份可用, 数据就是可用的。

(3) 效率和灵活性

分布式系统分散了工作负荷, 缓解了单机容量的压力。数据可以存储在临近的常用结点, 如果本结点的数据子集包含了要查询的全部内容, 显然会比集中式数据库在全集上查找节省时间。

系统易于实现扩展。例如, 一个单位要增加新的机构, 分布式数据库系统能够在对现有系统影响较小的情况下实现扩充。因此, 扩大系统规模比集中式系统更加方便、经济和灵活。

2. 数据库的发展方向

目前, 数据库技术的发展方向主要有两个: 一是改造和扩充关系数据库, 以适应新的应用要求; 二是改用新的数据库模型。目前, 这两个方面都取得了很大发展, 主要出现了下列新的数据库技术。

(1) 并行数据库技术

并行数据库技术包括对数据库的分区管理和并行查询。它通过将一个数据库任务分割

成多个子任务的方法由多个处理机协同完成这个任务，从而极大地提高了事务处理能力，并且通过数据分区可以实现数据的并行 I/O 操作。

(2) 数据仓库和数据挖掘技术

所谓数据仓库就是按决策目标将传统的事务型数据库中的数据重新组织划分，由此造成一种面向主题的、集成的、稳定的及随时间发展的数据集合。数据仓库与传统数据库的区别在于存储的数据容量大、存储的数据时间跨度大、存储的数据来源复杂、可用于企业与组织的决策分析处理等。所谓数据采矿(Data Mining, DM)就是从大型数据库或数据仓库的数据中提取人们感兴趣的、隐含的、事先未知的、潜在的知识。数据采矿方法的提出使人们有能力从过去若干年时间里积累的海量的、以不同的形式存储的、十分繁杂的数据资料中认识数据的真正价值。目前，数据采矿的研究已与数据库的研究结合起来。

(3) 多媒体数据库

人们通常把能够管理数值、文本、图形、图像和声音等媒体类型的数据库称为多媒体数据库(Multimedia Database)。与传统数据库管理系统(DBMS)一样，多媒体 DBMS 也要进行数据的处理、查询和事物的管理等，但是多媒体数据库有不同的用户接口和存储构造，在多媒体 DBMS 中特别强调“媒体独立性”(所谓媒体独立性是指不论管理的多媒体数据的媒体如何变化，都不需改变 DBMS)。

(4) 模糊数据库

传统的数据库仅允许对精确的数据进行存储和处理，而客观世界中许多事物是不精确的。模糊数据库技术的研究和实践就是为了解决模糊数据的表达和处理问题，使得数据库描述的模型更自然、更贴切地反映客观世界。

(5) 网络数据库

网络数据库是数据库技术与 Web 技术相互融合的技术。

此外，还有其他一些新的数据库技术，如：模糊演绎数据库、主动数据库、集中式工程数据库和面向对象数据库等。

1.2 数据模型

模型是现实世界特征的模拟和抽象。数据模型也是一种模型，只不过它模拟的对象是数据。根据模型应用的不同层次和目的，可以将模型分为两类，第一类是概念模型，按用户的观点来对数据和信息建模，主要用于数据库设计；另一类数据模型，主要包括网状模型、层次模型和关系模型等，它是按计算机系统的观点对数据建模。

1.2.1 数据模型的概念

数据模型是现实世界数据特征的抽象。数据模型是工具，是用来抽象、表示和处理现实世界中的数据和信息的工具。在数据库用数据模型这个工具来抽象、表示和处理现实世界的数据和信息，现有数据库系统均是基于某种数据模型的。

数据模型应满足三个方面的要求：

- (1) 能够比较真实地模拟现实世界;
- (2) 容易被人理解;
- (3) 便于在计算机系统中实现。

1.2.2 数据模型的组成要素

数据模型是由数据结构、数据操作和数据的约束条件三部分组成的。

数据结构是所研究对象的集合,这些对象是数据库的组成成分,例如表中的字段、名称等。数据结构分为两类,一类是与数据类型、内容、性质有关的对象;一类是与数据之间联系有关的对象。

数据操作是指对数据库中各种对象(型)的实例(值)允许执行的操作的集合,包括操作及有关的操作规则。数据库的操作主要有检索和更新两大类。数据模型必须定义数据操作的确切含义、操作符号、操作规则以及实现操作的语言。

数据的约束条件是一组完整性规则的集合。完整性规则是给定的数据模型中数据及其联系所具有的制约和依存规则,用以限定符合数据模型的数据库状态以及状态的变化,以保证数据的正确、有效和相容。

数据模型给出了在计算机系统上描述和动态模拟现实数据及其变化的一种抽象方法,数据模型不同,描述和实现方法亦不相同,相应的支持软件,即数据库管理系统也就不同。严格地讲,一个数据模型应由上述三部分组成,但数据结构是数据模型的本质标志。

1.2.3 概念模型

概念模型是现实世界到信息世界的第一层抽象,是现实世界到计算机的一个中间层次。概念模型是数据库设计的有力工具和数据库设计人员与用户之间进行交流的语言。它必须具有较强的语义表达能力,能够方便、直接地表达应用中的各种语义知识,且简单、清晰、易于用户理解。

在现实世界中,事物之间的联系是客观存在的。概念世界是现实世界在人们头脑中的反映,是对客观事物及其联系的一种抽象描述,不是现实世界的简单录像,而是把现实世界中的客观对象抽象为某一种信任结构,这种信任结构不是某一个DBMS支持的数据模型,而是概念级的模型。

建立概念模型涉及以下几个术语:

(1) 实体(entity)

客观存在并可相互区别的事物称为实体。实体可以是实际事物,也可以是抽象事件。比如,一个职工、一个部门属于实际事物;一次定货、借阅若干本图书、一场演出是比较抽象的事件。

同一类实体的集合称为实体集。例如,全体学生的集合、全馆图书等。我们用命名的实体型表示抽象的实体集,实体型“学生”表示全体学生的概念,并不具体指职工甲或学生乙。

(2) 属性(attribute)

描述实体的特性称为属性。例如，学生实体用若干个属性(学号、姓名、性别、出生日期、籍贯等)来描述。属性的具体取值称为属性值，用以刻画一个具体实体。

(3) 关键字

如果某个属性或属性组合能够唯一地标识出实体集中的各个实体，可以选作关键字，也称为码。

(4) 联系(relationship)

实体集之间的对应关系称为联系，它反映现实世界事物之间的相互关联。联系分为两种，一种是实体内部各属性之间的联系，另一种是实体之间的联系。

(5) E-R 图

概念模型的表示方法有很多，常用实体-联系方法(E-R 方法或 E-R 图)来描述现实世界的概念模型，E-R 方法也称为 E-R 模型。

E-R 图有 3 个要素：

实体——用矩形并在框内标注实体名称来表示；

属性——用椭圆形表示，并用连线将其与相应的实体连接起来；

联系——用菱形表示，菱形框内写明联系名，并用连线分别与有关实体连接起来，同时在连线上标上联系的类型(1:1、1:n 或 m:n)。图 1.2 为 E-R 图的示例。

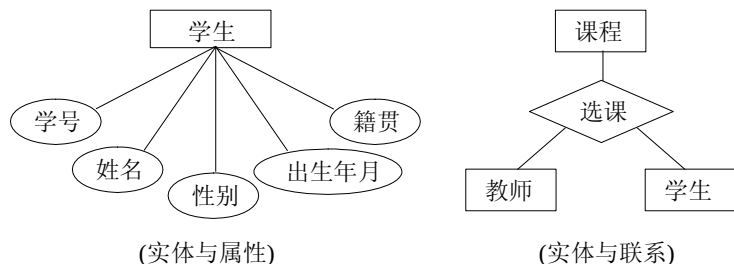


图 1.2 E-R图示例

联系有三种类型：

(1) 一对一联系

如果对于实体集 A 中的每一个实体，实体集 B 中至多有一个实体与之联系，反之亦然，则称实体集 A 与实体集 B 具有一对一联系，记为 1:1。

(2) 一对多联系

如果对于实体集 A 中的每一个实体，实体集 B 中有 n 个实体($n \geq 0$)与之联系，反之，对于实体集 B 中的每一个实体，实体集 A 中至多只有一个实体与之联系，则称实体集 A 与实体集 B 有一对多联系，记为 1:n。

(3) 多对多联系(m:n)

如果对于实体集 A 中的每一个实体，实体集 B 中有 n 个实体($n \geq 0$)与之联系，反之，对于实体集 B 中的每一个实体，实体集 A 中也有 m 个实体($m \geq 0$)与之联系，则称实体集 A 与实体集 B 具有多对多联系，记为 m:n。

1.2.4 常用的数据模型

每个数据库管理系统都是基于某种数据模型的。在目前数据库领域中，常用的数据模型有4种：层次模型、网状模型、关系模型和面向对象模型。

1. 层次模型

层次和网状模型是最早用于数据库系统的数据模型。层次模型的基本数据结构是层次结构，也称树型结构，树中每个结点表示一个实体类型，这些结点应满足：

- (1) 有且只有一个结点无双亲结点，这个结点称为根结点；
- (2) 其他结点有且仅有一个双亲结点。

在层次结构中，每个结点表示一个记录类型(实体)，结点之间的连线(有向边)表示实体间的联系。现实世界中许多实体间存在着自然的层次关系，如组织机构、家庭关系和物品分类等。图 1.3 就是一个层次模型的例子。

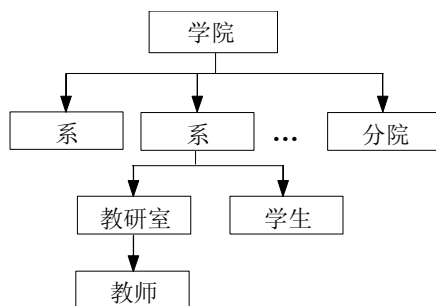


图 1.3 层次模型示例

2. 网状模型

网状模型的数据结构是一个网络结构。在数据库中，把满足以下两个条件的基本层次联系集合称为网状模型：

- (1) 一个结点可以有多个双亲结点；
- (2) 多个结点可以无双亲结点。

在网状模型中每个结点表示一个实体类型，结点间的连线表示实体间的联系。与层次模型不同，网状模型中的任意结点间都可以有联系，适用于表示多对多的联系，因此，与层次模型相比网状模型更具有普遍性。

网状模型虽然可以表示实体间的复杂关系，但它与层次模型没有本质的区别，它们都用连线表示实体间的联系，在物理实现上也有许多相同之处，如都用指针表示实体间的联系。层次模型是网状模型的特例，它们都称为格式化的数据模型。图 1.4 就是一个网状层次模型的例子。

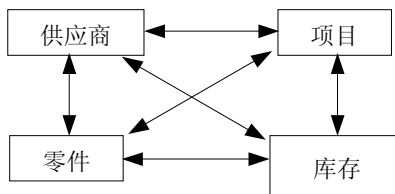


图 1.4 网状模型示例

3. 关系模型

关系模型的数据结构是二维表，由行和列组成。一张二维表称为一个关系。

图 1.5 是一个表示学生和教师任课情况的关系模型，图 1.5(a)和图 1.5(b)分别表示学生关系和教师任课关系。这两个关系也表示了学生和任课教师间的多对多联系，他们之间的联系是由在两个关系中的同名属性“班级”表示的。

学号	姓名	班级	...
30203301	杜少华	1	...
30203302	张大伟	1	...
30203303	王朋立	1	...
30203322	王小宁	1	...
30203327	林李鸣	1	...
...	...	...	...

(a) 学生关系

教师姓名	系别	任课名称	班级	...
吴丰	数学系	离散数学	1	...
赵立伟	外语系	英语	1	...
李健	计算机系	数据库技术	1	...
曲立涛	计算机系	操作系统	2	...
...	...	...	...	...

(b) 教师任课关系

图 1.5 学生与教师任课情况的关系模型

关系模型中的主要概念有：关系、属性、元组、域和关键字等。

与层次和网状模型相比，关系模型有下列优点：

- (1) 数据结构单一，不管实体还是实体间的联系都用关系来表示；
- (2) 建立在严格的数学概念基础上，具有坚实的理论基础；
- (3) 将数据定义和数据操纵统一在一种语言中，使用方便，易学易用。

4. 面向对象模型

面向对象的数据模型中的基本数据结构是对象，一个对象由一组属性和一组方法组成，属性用来描述对象的特征，方法用来描述对象的操作。一个对象的属性可以是另一个对象，另一个对象的属性还可以用其他对象描述，以此来模拟现实世界中的复杂实体。

在面向对象的数据模型中对象是封装的，对对象的操作通过调用其方法来实现。面向对象数据模型中的主要概念有：对象、类、方法、消息、封装、继承和多态等。

面向对象的数据模型有许多优点，主要有：

- (1) 可以表示复杂对象，精确模拟现实世界中的实体；
- (2) 模块化的结构，便于管理和维护；
- (3) 具有定义抽象数据类型的能力。

面向对象的数据模型是新一代数据库系统的基础，是数据库技术发展的方向。

1.2.5 数据库的体系结构

数据库系统的体系结构是数据库系统的一个总的框架。尽管实际的数据库系统的软件产品多种多样(支持不同的数据模型，使用不同的数据库语言，建立在不同的操作系统之上、数据的存储结构也各不相同)，但绝大多数的数据库系统在总的体系结构上都具有三级模式结构，即模式、外模式和内模式，如图 1.6 所示。

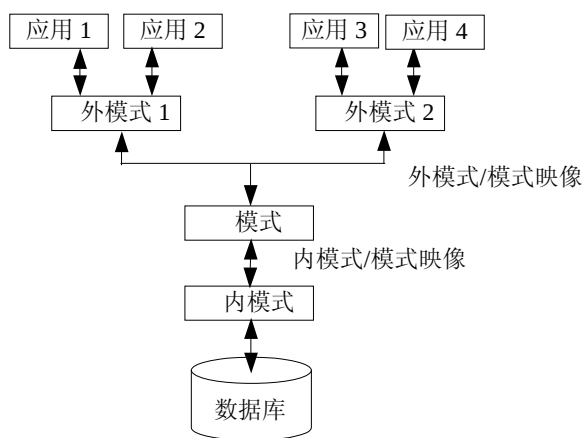


图 1.6 数据库系统的三级模式结构与二级映像

1. 数据库系统的三级模式机构

模式是数据库中全体数据的逻辑结构和特征的描述，模式与具体的数据值无关，也与具体的应用程序、高级语言以及开发工具无关，模式是数据库数据在逻辑上的视图。数据库的模式是唯一的，数据库模式是以数据模型为基础，综合考虑所有用户的需求，并将这些需求有机地结合成一个逻辑整体。

外模式也被称作用户模式，是用户和程序员最后看到并使用的局部数据逻辑结构和特征，一个数据库可以有若干个外模式。

内模式也称存储模式，是数据物理结构和存储方式的描述，是数据在数据库内部的保存方式，例如，数据保存在磁盘、磁带还是其他存储介质上；索引按照什么方式组织；数据是否压缩存储，是否加密等，内模式是物理的存储结构。

2. 数据库系统的二级映像功能

在数据库系统中，为实现在三级模式层次上的联系与转换，数据库管理系统在三级模

式之间提供了两层映像功能，这两层映像保证了数据库系统中的数据具有较高的逻辑独立性和物理独立性。

(1) 外模式/模式映像

模式描述数据的全局逻辑结构，外模式描述数据的局部逻辑结构。对应于一个模式可以有任意多个外模式。对应于每一个外模式，数据库系统都有一个外模式/模式映像，它定义了外模式与模式之间的映像对应关系。

应用程序是依据数据的外模式编写的，当数据库模式改变时，通过对各个外模式/模式的映像作相应改变，可以使外模式保持不变，从而不必修改应用程序，保证了数据与程序的逻辑独立性，简称数据的逻辑独立性。

(2) 内模式/模式映像

由于数据库只有一个模式，也只有一个内模式，所以数据库中的内模式/模式映像是唯一的。内模式/模式映像定义了数据全局逻辑结构与存储结构之间的对应关系。

当数据库的存储结构发生改变时(例如选用了另一种存储结构)，数据库管理员通过修改内模式/模式映像，可使模式保持不变，使应用程序不受影响，保证了数据与程序的物理独立性(简称数据的物理独立性)。

1.3 数据库设计

数据库设计是针对某个具体的应用问题进行信息抽象，构造概念模型，设计最佳的数据库结构，建立数据库及其应用系统的过程。

在现实世界中，信息结构十分复杂，应用领域千差万别，所以数据库的设计方法也多种多样。尽管如此，数据库的设计过程大体可以分为如下几个基本步骤。

(1) 需求分析阶段

这一阶段由系统分析员和用户双方共同收集数据库所需要的信息内容和用户对信息处理的需求，并以需求说明书的形式确定下来，作为以后系统开发的指南和系统验证的依据。需求分析阶段是数据库设计的基础，这一初始阶段做得好与坏，决定了后续设计的质量和速度，制约数据库应用系统设计的全过程。

(2) 概念设计阶段

数据库概念设计阶段是设计数据库的整体概念结构，也就是把用户需求统一到一个整体的逻辑结构中。概念设计的目标是产生反映用户需求的概念结构，即概念模型。概念模型能充分反映现实世界中实体间的联系，是各种基本数据模型的共同基础，同时也容易向现在普遍使用的关系模型转换。

(3) 逻辑设计阶段

概念设计的结果是得到一个与 DBMS 无关的概念模型。而逻辑设计的目的是把概念设计阶段设计好的全局概念模型转换成与选用的具体机器上的 DBMS 所支持的数据模型相符合的逻辑结构。对于逻辑设计而言，应首先选择 DBMS，但数据库设计人员往往没有挑选的余地，都是在指定的 DBMS 上进行逻辑结构的设计。

(4) 物理设计阶段

对于给定的基本数据模型，选取一个最适合应用环境的物理存储结构的过程，称为物

理设计。数据库的物理结构主要指数据库的存储记录格式、存储距离安排和存取方法。

(5) 实施阶段

数据库实施阶段是根据物理设计阶段的结果, 建立一个具体的数据库, 将原始数据载入到数据库中, 并编写应用系统程序, 对数据库进行试运行操作。

(6) 数据库的运行与维护阶段

在数据库试运行结果符合设计目标后, 数据库就可以真正投入运行了。数据库投入运行标志开发任务的基本完成和维护工作的开始。但这并不意味着设计过程的终结。由于应用环境在不断变化, 数据库运行过程中物理存储也会不断变化, 所以对数据库设计进行评价、调整和修改等维护工作是一个长期的任务, 也是设计工作的继续和提高。

1.4 Access 2003 数据库简介

1.4.1 Access的发展过程

1992 年 Microsoft 首次发布了用于 Windows 操作系统的第一个桌面关系型数据库管理系统, (Relational Date Base Management System, RDBMS)Access 1.0, 使 Access 逐步成为桌面数据库领域的领导者, 成为许多数据库管理用户的首选工具。1995 年末, Access 95 发布, 这是世界上第一个 32 位 RDBMS, 使得 Access 在桌面关系型数据库领域的应用得到普及和继续发展。1997 年 1 月, Access 97 发布, 使得 Access 数据库和 Web 得到最佳组合, 使企业能够通过 Intranet 方便地共享静态和动态的数据, 开拓了 Access 数据库从桌面向网络化发展的新天地。

伴随着新世纪的到来, Microsoft 发布了 Access 2000, 这已是 Microsoft 强大的桌面数据库管理系统的第六代产品, 是 32 位 Access 的第三个版本。Access 2000 与 Office 应用程序高度集成, 熟悉的界面使得迅速启动和运行更加容易。Access 2002 是微软公司办公自动化套装软件 Office XP 的组件之一, 对 Access 2002 的功能进行了重大改进和扩充。特别是与 Web 页面无缝集成, 从而能通过 Internet 共享数据信息。随 Office 2003 (又称 Office System) 一起发布的 Access 2003, 具有较 Access 2002 更强的组织、管理和共享数据库的能力。并且新增了导入、导出和处理 XML 数据文件的功能, 为用户提供了一个数据库管理工具集和数据库应用系统开发的理想环境。

1.4.2 Access 2003 新增功能

(1) 查看相关性信息

在 Access 2003 中, 可以查看数据库对象之间的相关性信息, 有助于随时对数据库进行维护。例如, 现在已不再需要“学生管理”数据库中的“按班级查询学生信息”查询, 但在删除前, 可能希望知道数据库中还有哪些对象在使用该查询, 从而, 在删除“按班级查询学生信息”查询前, 可以更改相关对象的记录源, 也可以删除它们。查看相关对象的完整列表, 有助于节省时间并减少错误。

除查看那些绑定到选中对象的对象列表外,还可以查看那些正由选定对象使用的对象。但不能对宏、模块和数据访问页进行相关性搜索, Access 项目不支持这项功能。

(2) 窗体和报表中的错误检查

在 Access 2003 中,可以启用自动错误检查以检查窗体和报表的常见错误。错误检查可指出错误,例如两个控件使用了同一键盘快捷方式,报表的宽度大于打印页面的宽度等。启用错误检查可帮助识别错误并更正它。

(3) 传播字段属性

在 Access 的早期版本中,只要修改了字段的被继承属性,就必须手动修改各个窗体和报表中相应控件的属性。现在,修改“表”设计视图中的被继承字段属性时, Access 2003 将显示一个选项,此选项用于更新全部或部分绑定到该字段的控件属性。

(4) 智能标记

在 Access 2003 中,可以使用 SmartTags 属性将智能标记添加到数据库中的表、查询、窗体、报表或数据访问页中的任何字段或控件中。

使用智能标记可以执行通常需要打开其他程序才能完成的操作。通过设置“智能标记”属性可将智能标记添加到字段或控件中。添加完后,当激活字段或控件中的单元格时,将显示“智能标记操作”按钮。单击该按钮可看到使用智能标记采取的操作菜单。

(5) 备份数据库或项目

对当前的数据库或项目作较大的改动之前,可以先对其进行备份, **备份将保存在默认的备份位置,或保存在当前文件夹中。**

若要恢复数据库,可以转到备份的位置,重命名该文件,然后在 Access 2003 中打开它。

(6) Windows XP 主题支持

Windows XP 操作系统提供了若干主题。如果选择了默认主题以外的主题,则 Access 2003 会将选中的主题应用到视图、对话框和控件。通过在数据库或项目中设置选项,可以防止窗体控件从操作系统中继承主题。

(7) 控件增强的排序功能

可以对窗体和报表的“列表框向导”和“组合框向导”以及 Access 2003 数据库的“查阅向导”中的最多 10 个字段指定升序或降序的排序方式。添加到这些向导的排序页看起来像“报表向导”中的排序页。

(8) 自动更正选项

在 Access 2003 中,可以更好地控制自动更正行为。“自动更正选项”按钮将出现在已自动更正的文本旁边。如果不希望更正该文本,则可以撤销更正,或者通过单击该按钮通过选择的方式来打开或关闭“自动更正”选项。

(9) SQL 视图中的增强字体功能和基于上下文的帮助

在 Access 2003 数据库、项目查询的 SQL 和查询“设计”视图中,可以使用新添加的“查询设计字体”选项(选择“工具”菜单下的“选项”命令,打开“选项”对话框的“表/查询”选项卡)更改文本的字体和字体大小。这些设置将应用到所有的数据库,而且可以使用计算机的高对比度和其他辅助功能设置。

在 Access 2003 数据库查询的 SQL 视图中,可以获得关于特定 Jet SQL 关键字、VBA 函数和 Access 函数的帮助,只需按 F1 键即可显示光标附近文本的相应帮助。此外,还

可以搜索 Jet SQL 和 VBA 函数的参考信息主题。

(10) XML 支持

XML(Extensible Markup Language, 可扩展标记语言) 是用于在网站上描述和传递数据的标准语言。Access 2003 提供导入和导出 XML 数据的方法, 还提供了使用 XML 相关文件与其他格式的数据进行相互转换的方法。

由于数据格式不同, 在 Internet 之间以及软件应用程序之间交换数据信息总是比较困难。虽然 HTML 适于为 Web 浏览器提供文本和图像信息, 但无法定义数据和数据结构。而 XML 则可以定义数据并说明数据的结构形式。与 HTML 相似, XML 也利用标记和属性, 但 HTML 指定了每个标记和属性的意义, 而 XML 使用标记仅用于分割数据片, 并将数据的解释完全留给要读取该数据的应用程序。

Access 2003 允许选择使用来自许多外部源的数据。几乎所有外部应用程序的数据在转换以用于 Access 2003 时, 借助 XML 就会容易实现。可以将任何 XML 数据转换为 Access 2003 支持的格式, 也可以指定是覆盖任何现有的表或是追加到现有的数据中。

在 Access 2003 中, 可以通过导出将数据输出到 XML 中。导出的对象可以是表、查询窗体或报表。

(11) 安全性增强

Access 2003 允许通过设置宏安全等级来防止可能不安全的 Visual Basic for Applications (VBA) 代码。可以设置安全等级, 以在每次打开包含 VBA 代码的数据库时接收提示, 也可以自动阻止那些来自于未知源的数据库。

另外, Access 2003 使用了 Microsoft Authenticode 技术, 此技术可使用数字证书来对宏项目进行数字签名, 用于创建此签名的证书将确认宏是否源于该签名者, 该签名将确认数字签名没被更改。设置宏安全等级时, 可以根据宏是否由可信任源列表上的开发人员进行数字签名来运行宏。

Access 2003 可利用 Microsoft Jet Expression Service 增强的沙盒模式来阻止表达式中所使用的潜在不安全的功能。

习 题

1. 解释下列术语:

数据库	数据库管理系统	数据库系统	模式
实体	属性	关键字	联系

2. 数据库系统的特点是什么?
3. 数据库的发展方向主要有哪些?
4. 数据模型的组成要素是什么?
5. 常用的数据模型有哪些? 各具有什么特点?
6. 什么是数据库的三级模式结构与二级映像?
7. 数据库的设计过程包括哪几个主要步骤?

第 2 章 关系数据库设计理论

本章学习目标

关系数据库是目前各类数据库中最重要、最流行的数据库。20 世纪 80 年代以来，计算机软件厂商新推出的数据库管理系统产品几乎都是关系数据库，非关系系统的产品也大都增加了关系接口。本章从关系模型入手，对关系数据结构、关系代数、关系完整性以及关系数据库的设计规范进行专门介绍。通过本章的学习，读者应该掌握以下内容：

- 关系模型的概念与分类
- 关系数据结构及定义
- 关系代数的基本运算
- 关系完整性
- 关系数据库设计与设计规范

关系数据库是支持关系模型的数据库系统，它应用数学的方法来处理数据库中的数据。系统而严格地提出关系模型的是美国 IBM 公司的 E.F.Codd，他于 1970 年发表的题目为“A Relational Model of Data for Shared Data Banks”的论文，奠定了关系数据库的理论基础。

目前，关系数据库已经成为计算机领域技术成熟、应用广泛的数据处理工具，以 SQL Server、Sybase 和 Oracle 为代表的关系数据库产品得到越来越广泛的应用。

2.1 关系模型

2.1.1 关于关系的一些术语

“关系”就是关系数据模型的数据结构，刻画关系数据结构就是要定义关系。从本质上来讲，关系是一个数学概念，具体说，是一个集合论中的概念，因此，从集合论的角度给出关系数据结构的形式化定义就是十分自然的事情。这样就将关系数据模型置于严格的数学基础之上。

(1) 关系

在关系模型中，一个关系就是一张二维表，每一个关系有一个关系名。在数据库中，一个关系存储为一个数据表。

(2) 属性

表中的列称为属性，每一个列有一个属性名，对应数据表中的一个字段。

(3) 元组

表中的行称为元组。一行就是一个元组，对应数据表中的记录，元组的各分量分别对应于关系的各个属性。关系模型要求每个元组的每个分量都是不可再分的数据项。

(4) 域

具有相同数据类型的值的集合称为域(Domain)，域是属性的取值范围，即不同元组对同一个属性的取值所限定的范围。

(5) 候选码

如果关系中的某个属性或属性组能唯一地标识一个元组，称该属性或属性组为候选码。

(6) 主码(主键)

若一个关系中有多个候选码，则选定其中一个为主码(主键)。主码的属性称为主属性。

(7) 外码(外键)

设F是基本关系R的一个或一组属性，但不是R的候选码，如果F与基本关系S的主码K_s相对应，则称F为基本关系R的外码。

2.1.2 关系模型

关系数据库采用关系数据模型(简称关系模型)作为数据组织方式，而关系模型是用二维表格结构来表示实体及实体间联系的模型。二维表中的每一列代表实体或实体间关系的某种属性，二维表中的一行叫做一个元组，是记录类型的实例，代表了某个具体的实体或具体实体间的特定关系。关系模型不仅可以方便地表示两个实体类型间的 1:1、1:n 关系，而且可以直接描述它们之间的 m:n 关系。

关系模型由关系数据结构、关系操作集合和完整性规则三部分组成。

关系模型的特点：

- (1) 关系必须规范化，指关系模型中的每一个关系模式都必须满足一定的要求；
- (2) 模型概念单一；
- (3) 集合操作，操作对象和结果都是元组的集合，即关系。

2.2 关系数据结构及形式化表示

在关系模型中，无论是实体还是实体之间的联系都由单一的结构类型关系来表示。

2.2.1 关系数据结构

(1) 笛卡儿积(Cartesian Product)

设有一组域 $D_1, D_2, \dots, D_n$ ，这些域可以部分或者全部相同。域 $D_1, D_2, \dots, D_n$ 的笛卡儿积定义为如下集合：

$$D_1 \times D_2 \times \dots \times D_n = \{(d_1, d_2, \dots, d_n) | d_i \in D_i, i=1, 2, \dots, n\}$$

其中每一个元素 $(d_1, d_2, \dots, d_n)$ 称为一个 n 元组(或简称元组)，元素中的每一个值 d_i 称为一个分量。

若干个域的笛卡儿积具有相当多的元素，在实际应用中可能包含许多“无意义”的元素。人们通常感兴趣的是笛卡儿积的某些子集，笛卡儿积的子集就是一个关系。

两个集合 R 和 S 的笛卡儿积是元素对的集合, 该元素对是通过选择 R 的某一元素(任何元素)作为第一个元素, S 的元素作为第二个元素构成的, 该乘积用 $R \times S$ 表示。笛卡尔积的结果可表示为一个二维表, 表中的每行对应一个元组, 表中的每列对应一个域。

例如, 我们给出三个域:

D_1 =导师集合 导师=张毅, 刘德成

D_2 =专业集合 专业=计算机专业, 通信专业

D_3 =研究生集合 学生=吕景刚, 王弼, 李兵

则 D_1 , D_2 , D_3 的笛卡尔积为:

$D_1 \times D_2 \times D_3 = \{(张毅, 计算机专业, 吕景刚), (张毅, 计算机专业, 王弼), (张毅, 计算机专业, 李兵), (张毅, 通信专业, 吕景刚), (张毅, 通信专业, 王弼), (张毅, 通信专业, 李兵), (刘德成, 计算机专业, 吕景刚), (刘德成, 计算机专业, 王弼), (刘德成, 计算机专业, 李兵), (刘德成, 通信专业, 吕景刚), (刘德成, 通信专业, 王弼), (刘德成, 通信专业, 李兵)\}$

该笛卡儿积的基数为 $2 \times 2 \times 3 = 12$, 这也就是说 $D_1 \times D_2 \times D_3$ 一共有 $2 \times 2 \times 3 = 12$ 个元组, 这 12 个元组的总体可组成一张二维表, 如表 2.1 所示。

表 2.1 D_1 , D_2 , D_3 的笛卡儿积

导 师	专 业	学 生
张 毅	计算机专业	吕景刚
张 毅	计算机专业	王 弼
张 毅	计算机专业	李 兵
张 毅	通信专业	吕景刚
张 毅	通信专业	王 弼
张 毅	通信专业	李 兵
刘德成	计算机专业	吕景刚
刘德成	计算机专业	王 弼
刘德成	计算机专业	李 兵
刘德成	通信专业	吕景刚
刘德成	通信专业	王 弼
刘德成	通信专业	李 兵

(2) 关系(Relation)

笛卡儿积 $D_1 \times D_2 \times \cdots \times D_n$ 的子集 R 称为在域 $D_1 \times D_2 \times \cdots \times D_n$ 上的一个关系(Relation), 通常表示为:

$$R(D_1, D_2, \cdots, D_n)$$

其中, R 表示关系的名称, n 称为关系 R 的元数或度数(Degree), 而关系 R 中所含有的元组个数称为 R 的基数(Cardinal Number)。

关系是笛卡儿积的子集, 所以关系也是一个二维表, 表的每行对应一个元组, 表的每列对应一个域。由于域可以相同, 为了加以区分, 必须为每列起一个名字, 称为属性

(attribute), N 目关系必有 n 个属性。

例如, 可以在表 2.1 的笛卡儿积中取出一个子集来构造一个关系。由于一个研究生只师从于一个导师, 学习某一个专业, 所以笛卡儿积中的许多元组是无实际意义的, 从中取出有实际意义的元组来构造关系。给关系命名为 SAP, 属性名就取域名, 即导师、专业和学生, 则这个关系可以表示为: SAP(导师, 专业, 学生)。

假设导师与专业是一对一的, 即一个导师只有一个专业; 导师与研究生是一对多的, 即一个导师可以带多名研究生, 而一名研究生只有一个导师, 这样 SAP 关系可以包含 3 个元组, 如表 2.2 所示。

表 2.2 SAP 关系

导 师	专 业	学 生
张 毅	通信专业	吕景刚
张 毅	通信专业	王 强
刘德成	通信专业	李 兵

假设学生不会重名(这在实际当中是不合适的, 这里只是为了举例方便), 则“学生”属性的每一个值都能唯一地标识一个元组, 因此可以作为 SAP 关系的主码。

关系可以有三种类型: 基本关系(通常又称为基本表或基表)、查询表和视图表。基本表是实际存在的表, 它是实际存储数据的逻辑表示; 查询表是查询结果对应的表; 视图表是虚表, 是由基本表或其他视图表导出的表, 不对应实际存储的数据。

由上述定义可以知道, 域 $D_1, D_2, \dots, D_n$ 上的关系 R , 就是由域 $D_1, D_2, \dots, D_n$ 确定的某些元组的集合。

在关系模型中, 对关系作了下列规范性限制:

- ① 关系中不允许出现相同的元组;
- ② 不考虑元组之间的顺序, 即没有元组次序的限制;
- ③ 关系中每一个属性值都是不可分解的;
- ④ 关系中属性顺序可以任意交换;
- ⑤ 同一属性下的各个属性的取值必须来自同一个域, 是同一类型的数据;
- ⑥ 关系中各个属性必须有不同的名字。

2.2.2 关系模式

关系模式是对关系的描述。关系是元组的集合, 也就是笛卡儿积的一个子集。因此关系模式必须指出这个元组集合的结构, 即它由哪些属性构成, 这些属性来自哪些域, 以及属性和域之间的映像关系。

其次, 一个关系通常是由赋予它的元组语义来确定的, 元组语义实质上是一个 n 目谓词(n 是属性集中属性的个数)。凡使该 n 目谓词为真的笛卡儿积中的元素(或者说凡符合元组语义的元素)的全体就构成了该关系模式的关系。现实世界随着时间在不断地变化, 因而在不同的时刻, 关系模式的关系也会有所变化。但是, 现实世界的许多已有事实限定关系模式所有可能的关系必须满足一定的完整性约束条件。这些约束或者通过对属性取值范围

的限定(例如,职工年龄小于 65 岁)或者通过属性值间的相互关联(主要体现在值的相等与否)反映出来,关系模式应当刻画出完整性约束条件。因此一个关系模式应当是一个五元组。

关系的描述称为关系模式(relation schema)。它可以形式化地表示为:

$$R(U, D, \text{Dom}, F)$$

其中 R 为关系名, U 为组成关系的属性名集合, D 为属性组 U 中属性所来自的域, Dom 为属性向域的映像集合, F 为属性间数据依赖关系的集合。

在关系数据库中,关系模式是型,关系是值,关系模式是静态的,关系是关系模式在某一时刻的状态或内容,关系是动态的。

2.3 关系代数

关系数据操作就是关系的运算。关系的基本运算有两类:传统的集合运算(并、交、差等)和专门的关系运算(选择、投影、联接),关系数据库进行数据查询时有时需要几个基本运算的组合。

2.3.1 集合运算

并、差、交是集合的传统运算形式,进行集合运算的关系 R 与 S 必须具有相同的模式,即 R 和 S 必须具有相同的属性集。

1. 集合的并运算

设有关系 R、S(R、S 具有相同的模式),二者的“并”运算定义为:

$$R \cup S = \{t | t \in R \vee t \in S\}$$

式中“ $\cup$ ”为并运算符, t 为元组变量,结果 $R \cup S$ 为一个新的与 R、S 同类的关系,该关系是由属于 R 和 S 的元组构成的集合。

例如,合并两个相同结构的数据表,就是两个关系的并集。

2. 集合的差运算

设有关系 R、S(R、S 具有相同的模式),二者的“差”运算定义为:

$$R - S = \{t | t \in R \wedge t \notin S\}$$

式中“-”为差运算符, t 为元组变量,结果 $R - S$ 为一个新的与 R、S 同类的关系,该关系是由属于 R 但不属于 S 的元组构成的集合,即在 R 中减去与 S 中元组相同的那些元组。

例如,设有选修 C 语言的学生关系 R,选修计算机基础的学生关系 S。查询选修了 C 语言而没有选修计算机基础的学生,就可以使用差运算。

3. 集合的交运算

设有关系 R、S(R、S 具有相同的模式),二者的“交”运算定义为:

$$R \cap S = \{t | t \in R \wedge t \in S\}$$

式中“ $\cap$ ”为交运算符, 结果 $R \cap S$ 为一个新的与 R 、 S 同类的关系, 该关系是由属于 R 而且属于 S 的元组构成的集合, 即两者所有的相同的那些元组的集合。

例如, 设有选修 C 语言的学生关系 R , 选修计算机基础的学生关系 S 。要查询既选修了 C 语言又选修计算机基础的学生, 就可以使用交运算。

2.3.2 专门的关系运算

在 Access 2003 数据库中, 查询是经常使用的数据操作, 学习专门的关系运算有助于查询的设计。

1. 选择运算

选取关系中满足一定条件的元组。

选择运算的形式定义为: 设有关系 R , 选择条件(逻辑表达式)用 F 表示, 则从关系 R 中选择出满足条件 F 的元组定义为:

$$\sigma_F(R) = \{ t | t \in R \wedge F(t) = \text{true} \}$$

例如, 从学生表中查询系别为“计算机系”的学生信息, 使用的查询操作就是选择运算。

2. 投影运算

选取关系中的某些列, 并且将这些列组成一个新的关系。

投影运算的形式定义为: 设有关系 R , 其元组变量为 $t^k = \langle t_1, t_2, \dots, t_k \rangle$, 那么关系 R 在其分量 $A_{i1}, A_{i2}, \dots, A_{in}$ ($n \leq k, i1, i2, \dots, in$ 为 1 到 k 之间互不相同的整数) 上的投影定义为:

$$\prod_{i1, i2, \dots, in}(R) = \{ t | t = \langle t_{i1}, t_{i2}, \dots, t_{in} \rangle \wedge \langle t_1, t_2, \dots, t_k \rangle \in R \}$$

例如, 从学生表中查询学生的学号和姓名信息, 使用的查询操作就是投影运算。

3. 联接运算

用笛卡儿积可以建立两个关系间的联接, 但建立的关系是一个较为庞大的体系, 而且不符合实际操作的需要。在实际问题当中, 两个关系相互联接一般是满足某些条件的, 所得到结果往往比较简单。因此, 对于笛卡儿积可以做适当的限制, 以适应实际应用的需要, 这样就引入了联接运算(又称为 θ 联接)。

联接运算将两个关系拼接成一个更宽的关系, 新关系中包含满足联接条件的元组。

联接运算的形式定义为: 设有关系 R 、 S , 同时 $i\theta j$ 是一个比较式, 其中 i 、 j 分别为 R 和 S 中的域, θ 为算术比较符, 此时关系 R 、 S 在域 i 、 j 上的 θ 联接定义为:

$$R \underset{i\theta j}{><} S = \sigma_{i\theta j}(R \times S)$$

式中“ $><$ ”为联接运算符。该式说明, R 与 S 的 θ 联接是 R 与 S 的笛卡儿积再加上限制 $i\theta j$ 而成, 显然, $R \underset{i\theta j}{><} S$ 中元组的个数远远少于 $R \times S$ 的元组个数。

联接运算有多种类型，自然联接是最常用的联接运算。在联接运算中，按关系的属性值对应相等为条件进行的联接操作称为等值联接，自然联接是去掉重复属性的等值联接。

2.4 关系完整性

关系模型的完整性规则是对关系的某种约束条件。

关系模型有三类完整性约束：实体完整性、参照完整性和用户定义的完整性。其中实体完整性和参照完整性是关系模型必须满足的完整性约束条件，被称作是关系的两个不变性，应该由关系系统自动支持。

1. 实体完整性(Entity Integrity)

实体完整性规则：若属性 A 是基本关系 R 的主码，则属性 A 不能取空值。

例如，在关系学生(学号，姓名，性别，年龄，专业，班级)中，“学号”属性为主码，则“学号”不能取空值。

实体完整性规则规定基本关系的所有主属性都不能取空值，而不仅是主码整体不能取空值。例如学生成绩关系(学号，课程号，成绩)中，“学号、课程号”为主码，则“学号”和“课程号”两个属性都不能取空值。

对于实体完整性规则说明如下：

(1) 实体完整性规则是针对基本关系而言的。一个基本表通常对应现实世界的一个实体集。例如学生关系对应于学生的集合。

(2) 现实世界中的实体是可区分的，即它们具有某种唯一性标识。例如学生关系中的“学号”，学生可以有重名的，但学号没有重复的。

(3) 相应地，关系模型中以主码作为唯一性标识。

(4) 主码中的属性即主属性不能取空值。所谓空值就是“不知道”或“无意义”的值。

2. 参照完整性(Referential Integrity)

现实世界中的实体之间往往存在某种联系，在关系模型中，实体及实体间的联系都是用关系来描述的，自然就存在着关系与关系之间的引用。

【例 2.1】 职工实体和车间实体可以用下面的关系表示，其中主码用下划线标识：

职工(职工号，姓名，性别，年龄，车间号，车间主任)

车间(车间号，车间名)

这两个关系之间存在着属性的引用，即职工关系引用了车间关系的主码“车间号”。显然，职工关系中的“车间号”属性取值，必须是车间关系确实存在的车间号。

【例 2.2】 学生、课程和成绩实体之间的多对多联系可以用如下三个关系表示：

学生(学号，姓名，性别，年龄，专业，班级)

课程(课程号，课程名，学分)

成绩(学号，课程号，成绩)

这三个关系之间也存在着属性的引用，成绩关系引用了学生关系的主码“学号”和课

程关系的主码“课程号”。成绩关系中的“学号”值必须是确实存在的学生的学号；成绩关系中的“课程号”值也必须是确实存在的课程的课程号。

外码、参照关系以及被参照关系

设 F 是基本关系 R 的一个或一组属性，但不是关系 R 的码。如果 F 与基本关系 S 的主码 K 相对应，则称 F 是基本关系 R 的外码(外部关键字, Foreign Key)，并称基本关系 R 为参照关系(Referencing Relation)，基本关系 S 为被参照关系(Referenced Relation)或目标关系(Target Relation)。关系 R 和 S 不一定是不同的关系。

显然，目标关系 S 的主关键字 K 和参照关系 R 的外码 F 必须定义在同一个(或同一组)域上。

在例 2.1 中，职工关系的“车间号”属性与车间关系的主码“车间号”相对应，因此“车间号”属性是职工关系的外码。这里的车间关系是被参照关系，职工关系是参照关系。在例 2.2 中，成绩关系的“学号”属性与学生关系的主码“学号”相对应，“课程号”属性与课程关系的主码“课程号”相对应，因此“学号”和“课程号”属性是课程关系的外码。这里学生关系和课程关系均为被参照关系，成绩关系为参照关系。

需要指出的是，外码不一定要与相应的主码同名，不过在实际应用当中，为了便于识别，当外码与相应的主码属于不同关系时，往往给它们取相同的名字。

参照完整性规则就是定义外码与主码之间的引用规则。

参照完整性规则：若属性(或属性组) F 是基本关系 R 的外码，它与基本关系 S 的主码 K 相对应(基本关系 R 和 S 不一定是相同的关系)，则 R 中的每个元组在 F 上的值必须为：

- (1) 或者取空值(F 的每个属性值均为空值)；
- (2) 或者等于 S 中的某个元组的主码值。

例如，职工关系中每个元组的“车间号”属性只能取下面两类值：

- (1) 空值，表示尚未给该职工分配车间；
- (2) 非空值，其取值必须是车间关系中某个元组的“车间号”值。

3. 用户定义的完整性(User-defined Integrity)

任何关系数据库系统都应该支持实体完整性和参照完整性。除此之外，不同的关系数据库系统根据其应用环境的不同，往往还需要一些特殊的约束条件，用户定义的完整性就是针对某一具体关系数据库的约束条件。它反映某一具体应用所涉及的数据必须满足的语义要求。例如某个属性的取值必须唯一、某些属性值之间应满足一定的函数关系、某个属性的取值范围在 $0 \sim 100$ 之间等。关系模型应提供定义和检验这类完整性的机制，以便用统一的方法处理它们。

2.5 数据库设计与设计规范

2.5.1 关系数据库设计规范

规范化理论是研究关系模式中各属性之间的依赖关系以及对关系模式性能的影响，探

讨论关系模式应该具备的性质和设计方法。

关系的规范化理论是由 E.F.Codd 于 1971 年系统提出的。规范化理论为数据结构定义了五种规范化模式,简称范式。它提供了判别关系模式设计的优劣标准,为数据库设计提供了严格的理论基础。

使用范式表示关系模式满足规范化的等级,满足最低要求的为第一范式(1NF),在第一范式中满足进一步要求的为第二范式,其余依次类推,规范化的进一步等级为 2NF、3NF。1974 年 Codd 和 Boyce 又共同提出了一个新范式,即 BCNF。1976 年 Fagin 又提出了 4NF,后来又有人提出了 5NF。通常只使用前三种范式。

2.5.2 第一范式(1NF)

第一范式(1NF): 在关系模式 R 中的每一个具体关系 r 中,如果每个属性值都是不可再分的最小数据单位,则关系 R 属于第一范式。

属于第一范式的关系应该满足的基本条件是元组中的每一个属性都不可分割。

例如,设计成绩关系如表 2.3 所示,由于成绩属性含有分项,不是不可再分的最小数据单位,所以,该成绩关系不符合第一范式。

表 2.3 成绩关系

学 号	课程号	成 绩			学 分
		实验成绩	期末成绩	综合成绩	
30203301	01	88	90	178	3
30203301	02	78	89	167	2
30203303	01	71	89	160	3
30203304	02	76	80	156	2
30203305	03	68	74	142	3
30203306	01	62	78	140	3

2.5.3 第二范式(2NF)

第二范式(2NF): 如果关系模式 $R(U, F)$ 中的所有非主属性都完全依赖于任意一个候选关键字(候选码),则关系 R 属于第二范式。

所谓第二范式,指的是一个关系不仅满足第一范式,而且所有非主属性完全依赖于其主码。

例如,关系成绩(学号,课程号,成绩,学分),其中,主码为组合关键字(“学号”、“课程号”),但学分不完全依赖于这个组合关键字,却完全依赖于课程号,这个关系就不符合第二范式。

使用以上关系模式会存在以下几个问题:

- (1) 数据冗余: 假设有 100 名学生选修同一门课,就要重复 100 次相同学分。
- (2) 更新复杂: 若调整了某门课程的学分,相应的记录(元组)学分值都要更新,不能

保证修改后的不同元组一门课学分完全相同。

(3) 插入异常：假如开一门新课，可能没有学生选修，没有学号关键字，只能等到有学生选修才能把课程号和学分加入。

(4) 删除异常：如果学生已经毕业，由于学号不存在，选修记录也必须删除。

产生以上问题的主要原因是：

非主属性“学分”仅依赖于课程号，不是完全依赖组合关键字(学号，课程号)。

解决的方法是将一个非 2NF 的关系模式分解为多个 2NF 的关系模式。

通过模式分解，将成绩关系分成两个关系模式，分别是关系成绩(学号，课程号，成绩)和关系课程(课程号，课程名称，学分)，新的成绩关系和课程关系之间通过成绩关系中的外码(外关键字)课程号与课程关系的课程号相联系，如表 2.4 所示。

分解后得到的成绩关系和课程关系满足第二范式(2NF)。

表 2.4 模式分解后的关系示例

30203301	01	88	178
30203301	02	78	167
30203303	01	71	160
30203304	02	76	156
30203305	03	68	142
30203306	01	62	140

(a)成绩关系

课程号	课程名称	学 分
01	计算机原理	3
02	计算机文化基础	2
03	数据库原理	3

(b)课程关系

2.5.4 第三范式(3NF)

第三范式(3NF)：如果关系模式 $R(U, F)$ 中的所有非主属性对任何候选关键字都不存在传递依赖，则关系 R 属于第三范式。

符合第三范式的关系不仅满足第二范式，而且它的任何一个非主属性都不传递依赖任何主关键字。

例如，有这样一个关系 $Student(\text{学号}, \text{姓名}, \text{性别}, \text{出生日期}, \text{院系}, \text{地址})$ ，关键字“学号”决定各个属性，由于是单一主码，没有部分依赖的问题。

但这个关系会存在大量的冗余数据，学生所在院系和地址属性是重复存储的信息，在插入、删除和修改时也将产生与上例类似的情况，也同样存在大量冗余和更新异常的问题。

数据冗余的原因是关系中存在传递依赖，因为学号 $\rightarrow$ 院系用“ $\rightarrow$ ”表示依赖关系，学号 $\rightarrow$ 院系表示院系依赖于学号成立，而院系 $\rightarrow$ 学号不成立，但院系 $\rightarrow$ 地址成立，因此学号

对地址的关系是通过传递依赖关系实现的，学号不直接决定非主属性地址。

通过模式分解，去掉传递依赖。把 Student 关系分解为两个关系 Stud(学号，姓名，院系编号)和 Sdep(院系编号，名称，地址)。

例如，在表 2.4(a)所示的成绩关系中，主码是学号和课程号，实验成绩、期末成绩依赖主码，但是综合成绩不依赖于主码，因此可以取消综合成绩属性，使成绩关系满足第三范式。

2.5.5 BC范式(BCNF)

在关系数据库中，除了函数依赖之外还有多值依赖、联结依赖的问题，这是第四范式所要规范的问题。

如果关系模式 $R(U, F)$ 的所有属性(包括主属性和非主属性)都不传递依赖于 R 的任何候选关键字，那么关系 R 属于 BC 范式。

例如，假设有关系成绩(学号，姓名，课程，成绩)。从关系中可以看出，课程和成绩不仅依赖于学号，还依赖于姓名，产生了多值依赖关系，关系不符合第四范式要求。通过模式分解，去掉多值依赖。将成绩关系分解为关系学生(学号，姓名)和关系成绩(学号，课程，成绩)，模式分解后得到的学生关系和成绩关系满足 BCNF。

如果关系模式不满足 BC 范式，同样会给插入和删除等操作带来麻烦。

2.5.6 关于规范化

一个低一级范式的关系模式，通过模式分解可以转化为若干个满足高一级范式的关系模式的集合，这一过程称为规范化。

(1) 规范化的目的

规范化的目的是使关系模式的结构更合理，消除存储异常，使数据冗余尽量小。便于插入、删除和更新，避免数据的不一致性，提高对关系的操作效率。

(2) 规范化原则

规范化的基本思想是逐步消除关系中数据依赖的不合适部分，使模式中的各关系模式达到某种程度的分离，实现一事一地的模式设计原则。因此，规范化实质是概念的单一化，是以一个关系模式描述一个实体或实体间的一种联系的原则来设计数据库。

(3) 规范化方法

关系模式的规范化是通过模式分解来实现的。将关系模式投影分解成两个或两个以上的关系模式。进行分解的目标是达到更高一级的规范化程度，但是分解的同时必须考虑两个问题：无损联接性和保持函数依赖。但不可能做到既无损联接，又完全保持函数依赖，应该根据实际需要进行权衡。

关系模式的分解方法并不是唯一的，最小冗余的要求必须以分解后的数据库能够表达原来数据库所有信息为前提来实现。实际上，根据关系的具体应用情况，例如，对于更新不频繁、查询频度极高的数据库系统来说，并不一定要求全部模式都达到 BCNF，保留部分冗余可能会方便数据查询。

习 题

1. 解释下列术语：
关系模型 关系模式 关系实例 属性 域 元组 主码
2. 传统的集合运算包括哪些？专门的关系运算有哪些？
3. 为什么关系中的元组没有先后顺序？
4. 为什么关系中不允许有重复的元组？
5. 关系与普通的表格、文件有什么区别？
6. 广义笛卡儿积、等值联接、自然联接三者之间有什么区别？
7. 设有关系 R、S 如下：

关系 R		
A	B	C
A	b	c
b	a	d
C	d	e
D	f	g

关系 S		
A	B	C
b	a	d
d	f	g

求 $R \cup S$ 、 $R - S$ 、 $R \cap S$ 、 $R \times S$ 。

8. 关系模型中包括哪三类完整性约束？
9. 简述 1NF 和 2NF 的主要内容。
10. 简述关系规范化的含义。

第 3 章 Access 2003 数据库

本章学习目标

Access 2003 是 Windows 环境下的数据库管理软件。它提供了大量的工具和向导，即使没有任何编程经验，也可以通过可视化的操作来完成大部分的数据库管理和开发工作。本章从 Access 2003 数据库的开发环境入手，介绍 Access 数据库的操作与管理以及 Access 数据库的安全性。通过本章学习读者应掌握以下内容：

- Access 2003 数据库的安装、启动与关闭
- Access 2003 数据库的构成
- 创建数据库
- 数据库管理
- 数据库的安全性

3.1 Access 2003 数据库开发环境

Access 2003 数据库管理系统可以管理从简单的文本、数字字符到复杂的图片、动画和音频等各种类型的数据。在 Access 2003 中，可以构造应用程序来存储和归档数据；并可以使用多种方式进行数据的筛选、分类和查询；还可以通过显示在屏幕上的窗体来查看数据；或者生成报表将数据按一定的格式打印出来，并支持通过 VBA 编程来处理数据库中的数据。

作为 Microsoft Office 2003 套件中的成员，Access 2003 的使用界面与 Word、Excel 等风格相同。

3.1.1 Access 2003 数据库的安装、启动与关闭

1. 安装

Access 2003 是一个 32 位的软件，可以运行在 Windows 9x/NT/2000/XP 等操作系统环境中。进入 Windows 系统，将 Office 2003 的安装光盘放入驱动器，稍后，系统会自动启动 Microsoft Office 2003 安装程序，根据窗口提示信息便可一步一步地完成 Access 2003 的安装。

如果已经安装了 Office 2003，一般来说 Access 2003 也会被安装，如果没有安装，也可以单独安装。使用“控制面板”窗口中的“添加/删除”按钮，可修复、删除 Access 2003，或者单独安装 Office 2003 组件。

2. 启动

启动 Access 2003 的方式,与启动其他 Office 软件完全一样,有通过开始菜单、桌面快捷方式等方式。无论是 Word、Excel、PowerPoint 还是 Access 2003, 它们的操作界面几乎都是相同的,差别只在于每一种软件有其专用的工具按钮或版式, Access 2003 的操作主界面如图 3.1 所示。

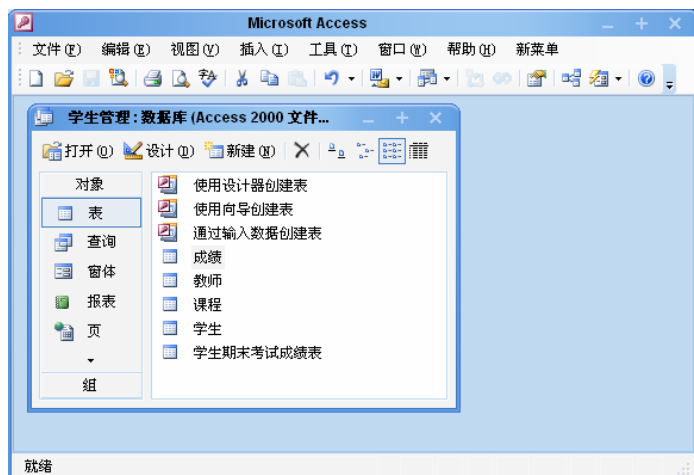


图 3.1 Access 2003 的操作主界面

3. 关闭

退出 Access 2003 的方法比较简单,可采用如下两种方法:

- (1) 选择“文件”菜单中的“退出”命令。
- (2) 单击 Access 标题栏右边的关闭按钮。

3.1.2 Access 2003 数据库构成

Access 2003 将数据库定义成一个.MDB 文件,并分成多个对象,在数据库窗口中的“对象”栏显示了数据库的 7 种对象:表、查询、窗体、报表、页、宏和模块,如图 3.1 所示。

1. 表

表(Table)是数据库最基本的组件,是存储数据的基本单元,由不同的列、行组合而成,每一列代表某种特定的数据类型,称之为字段,例如“学号”、“姓名”、“性别”等;每一行由各个特定的字段组成,称之为记录。

字段中存放的信息种类很多,包括文本、数字、日期、货币、OLE 对象(声音、图像等)以及超级链接等。每个字段包含了一类信息,大部分表中都要设置主码,以唯一地表示一条记录。在表内还可以定义索引,以加快查找速度。一个数据库中的多个表并不是孤立存在的,通过有相同内容的字段可在多个表之间建立关联。

例如,“学生”表的数据表视图如图 3.2 所示。

	学号	姓名	性别	出生日期	民族	身份证号码	婚否	政治面貌
+	30203301	杜少华	男	1985-10-10	汉族	37070619851010211X	☐	团员
+	30203302	张大伟	女	1986-2-14	汉族	37078519860214367x	☐	团员
+	30203303	王朋立	男	1984-6-1	汉族	371325198406011219	☐	团员
+	30203305	陈小东	男	1986-8-20	汉族	370126198608207138	☐	团员
+	30203306	王云	男	1984-6-19	汉族	370725198406190410	☐	党员
+	30203307	林冰冰	女	1984-1-19	汉族	370285198401192932	☐	团员
+	30203308	郑玉栋	男	1984-9-23	汉族	429006198409234274	☐	团员
+	30203310	朱云波	男	1984-6-11	汉族	372323198406111253	☐	团员
+	30203311	宋名	男	1983-1-10	汉族	370323198301108121	☐	团员
+	30203312	王开开	男	1984-8-20	汉族	370702198408203255	☐	团员
+	30203313	何云周	男	1985-1-10	汉族	430522198309106375	☐	团员
+	30203314	李明	男	1984-8-20	汉族	430524198503227122	☐	团员

图 3.2 “学生”表数据表视图

2. 查询

查询是通过设置某些条件，从表中获取所需要的数据。按照指定规则，查询可以从一个表、一组相关表和其他查询中抽取全部或部分数据，并将其集中起来，形成一个集合供用户查看。将查询保存为一个数据库对象后，可以在任何时候查询数据库的内容

3. 窗体

窗体是数据库和用户的一个联系界面，用于显示包含在表或查询中的数据和操作数据库中的数据。在窗体中，不仅可以包含普通的数据，还可以包含图片、图形、声音、视频等多种对象。当数据表中的某一字段与另一数据表中的多个记录相关联时，还可以通过主/子窗体进行处理。

例如，一个简单的查询窗体，如图 3.3 所示。

图 3.3 一个简单的查询窗体

4. 报表

报表通常作为数据统计的方式来使用，Access 2003 报表的设计与窗体类似，多用于按指定样式打印数据。利用报表也可以进行统计计算，如求和、求平均值等。

5. 页

页(或称为数据访问页)是自 Access 2000 之后新增的对象，指的就是网页(Web Page)。在 Access 2000 以前的版本中，在创建表、查询之后，以“另存为 Web 页”的方法产生网页，在 Access 2003 中，可以直接创建页对象。

6. 宏

宏(Macro)是由一系列命令组成的集合,以实现自动执行重复性工作的功能,例如设定打开 Access 2003 时自动打开某个窗体、表或报表等。使用宏可以简化一些经常性的操作。如果将一系列操作设计为一个宏,在执行宏时,其中定义的所有操作会按照规定的顺序依次执行。

宏可以是包含一系列操作的一个宏，也可以是由若干个宏组成的宏组。另外，还可以在宏操作中加条件来控制其是否执行。

7. 模块

模块(Module)是用 VBA 语言编写的程序段，它以 Visual Basic 为内置的数据库程序语言。对于数据库的一些较为复杂或高级的应用功能，需要使用 VBA 代码编程实现。通过在数据库中添加 VBA 代码，可以创建出自定义菜单、工具栏和具有其他功能的数据库应用系统。

模块由声明、语句和过程组成。Access 2003 有两种类型的模块：标准模块和类模块。

类模块属于一种与某一特定窗体或报表相关联的过程集合，这些过程均被命名为事件过程，作为窗体或报表处理某些事件的方法。

标准模块包含与任何其他对象都无关的常规过程，以及可以从数据库任何位置运行的经常使用的过程。标准模块和与某个特定对象相关的类模块的主要区别在于其范围和生命周期。

3.1.3 Access 2003 数据库操作环境

在如图 3.1 所示的 Access 2003 数据库主界面中，包括标题栏、菜单栏、工具栏、状态栏和数据库窗口等。其中菜单栏、工具栏的操作与 Word 和 Excel 完全相同。这里我们主要介绍数据库窗口。

1. 数据库窗口

在创建或打开 Access 2003 数据库后，所有的数据库操作都是围绕数据库窗口进行的，数据库窗口由对象选项卡、对象列表和工具栏组成。

数据库窗口左边的对象列表，列出了 Access 2003 中的 7 种标准对象种类。单击某个对象就可以进入相应的操作窗口，还可以通过“视图”菜单中的“数据库对象”命令，选择要显示的对象种类。

数据库窗口的主要界面是显示各个对象选项卡中包括的对象名称，如“表”选项卡中的“学生”、“课程”等，以及用于创建对象的快捷方式。数据库窗口工具栏中的按钮，则用于操作数据库窗口中的各种对象，“打开”、“设计”、“新建”按钮用于启动或打开选定的对象，其他按钮与 Word 中的按钮类似。

2. 组

除了标准的 7 种对象外，Access 2003 还允许根据用户需求创建新组，以便摆放数据库中常用对象的快捷方式。

右击某个对象，在弹出的菜单中选择“新组”命令，屏幕上会出现“新建组”对话框，如图 3.4 所示。



图 3.4 “新建组”对话框

在“新组名称”文本框中输入要创建的新组名称，如“学生管理”，单击“确定”按钮后，就会在组列表中添加一个新组“学生管理”。使用快捷菜单，还可以重命名、删除已有的组。

将一个数据库对象添加到组的方法是：右击该对象，在弹出菜单中选择“添加到组”子菜单中的相应分组。也可以将选中的某个数据库对象，直接拖动至某个组内，例如，将“表”对象中的“学生”拖至“学生管理”组中，则在“学生管理”中出现一个“学生”的快捷方式。

3. Access 2003 帮助系统

Access 2003 具有强大的帮助系统，全部采用 HTML 帮助的形式，通过 Microsoft Office Access 帮助系统，可以随时获得问题的答案。

“Office 助手”可以帮助用户查找“帮助”主题，显示提示针对用户操作的帮助信任。在默认情况下，Office 助手是被激活的，通过快捷菜单中的“隐藏”命令可以将其隐藏，使用“帮助”菜单中的“显示 Office 助手”命令可以重新显示 Office 助手。

当光标位于用户想获得帮助的选项处时，可以直接按下 F1 功能键，弹出一个帮助窗口，显示出关于该选项的信息。

如果计算机处于和互联网相连接的状态，还可以通过 Microsoft Office Online 命令，在 Microsoft 提供的 Web 站点上找到有用的信息以及最新的模板和向导。

3.2 Access 2003 数据库操作与管理

Access 2003 数据库管理系统将各种有关的表、索引、窗体、报表以及 VBA 程序代码都包含在一个 .MDB 文件中，并为用户处理了所有的文件管理细节。

建立一个 Access 2003 数据库的基本步骤为：

- (1) 进行系统分析，确定新建数据库的目的；
- (2) 设计数据库的表；
- (3) 规划表中的字段，并确定表的主码；
- (4) 确定表之间的关系；
- (5) 根据表间的关系，优化表、表中字段的设计；
- (6) 输入数据并创建其他数据库对象，如查询、窗体、报表、页、宏和模块等；

- (7) 使用数据库分析工具来分析和改进数据库的性能；
- (8) 设置数据库的安全性。

3.2.1 创建数据库

Access 2003 提供了两种创建数据库的方法：一是使用数据库向导来创建数据库，使用这种方法用户只需要做一些简单的选择操作，就可以建立相应的表、查询、窗体、报表和页等对象，从而建立一个完整的数据库；二是先创建一个空数据库，然后创建表、查询、窗体、报表和页等对象。无论使用哪一种方法，在创建数据库之后，都可以在任何时候修改或扩展数据库。

1. 使用向导创建数据库

如果需要在创建数据库的同时，就为所选的数据库创建所需的表、窗体及报表等对象，可以选择数据库向导来创建数据库。

具体操作可根据向导提示完成，在此不给出详细介绍。

2. 自定义创建数据库

常用创建数据库的方法是先创建一个空数据库，然后再创建其他数据库对象。

自定义创建数据库的步骤如下：

(1) 在“新建文件”任务窗格中，选择“空数据库”。

(2) 在弹出的“文件新建数据库”对话框中输入新建数据库文件的路径及名称，单击“创建”按钮便创建了一个空数据库。

Access 2003 在同一时间只能处理一个数据库，因而每新建一个数据库的同时，会自动关闭前面打开的数据库。

3. 数据库的打开与关闭

要打开一个已经存在的数据库，可以使用工具栏上的“打开”按钮，或“文件”菜单中的“打开”命令，在弹出的对话框中指定要打开的数据库文件即可。也可以使用“文件”菜单选择打开最近使用过的数据库。

在 Access 2003 中，数据库的打开有 4 种方式，如图 3.5 所示。

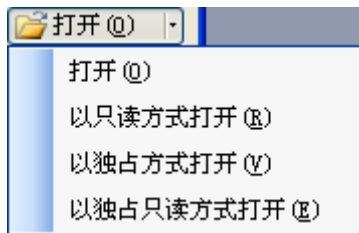


图 3.5 数据库的打开方式

- “打开”方式，以共享方式打开数据库文件。使用这种方式，网络上的其他用户可以再打开这个文件，也可以同时编辑这个文件，这是默认的打开方式。
- 以只读方式打开数据库文件。如果只是想查看已有的数据库而不想对它进行修改，

- 可以选择“以只读方式打开”，选择这种方式可以防止无意间对数据库的修改。
- 以独占方式打开数据库文件。可以防止网络上的其他用户同时访问这个数据库文件，也可以有效地保护自己对共享数据库文件的修改。
- 以独占只读方式打开数据库文件。不对数据库进行修改时，可以选择这种方式。选择这种方式可以防止网络上的其他用户同时访问这个数据。

为了方便打开文件，Access 2003 可以把数据库文件添加到收藏夹中，这样就可以快速打开数据库。使用视图菜单的“工具栏”中的 Web 命令，则屏幕上显示出 Web 工具栏，打开需要添加到收藏夹的数据库文件，单击 Web 工具栏的“收藏夹”按钮，从下拉列表中选择“添加到收藏夹”，打开添加收藏夹对话框，单击“添加”按钮即可。

如果要退出 Access 2003，只需要单击主窗口的关闭按钮，或者选择“文件”菜单中的“退出”命令。如果只想关闭数据库文件而不关闭 Access 2003，单击数据库窗口的关闭按钮即可。

3.2.2 数据库管理

在数据库应用系统的使用过程中，要保证数据的正确性、一致性，并使数据及时得到更新，数据库的管理是至关重要的。

1. 转换数据库

由于 Access 版本的不同，所创建的数据库应用系统的文件格式也会有所区别。在 Access 2003 中，可以将旧版本的 Access 数据库转换成新版本的数据库格式，也可以进行反向操作。

要转换一个数据库文件的格式，应先在 Access 2003 中打开此数据库，然后选择“工具”菜单中的“数据库实用工具”，在“转换数据库”子菜单中选择相应的选项。

Access 2003 是以“只读”方式读取旧版本(Access 2000 之前)的数据库，此模式不允许改变表的设计、窗体的对象设计等，除非将数据库整个转换为 Access 2003 的数据库格式，然后再将其保存。

2. 压缩和修复数据库

当删除窗体、报表等数据库对象时，Access 2003 不会释放它们占用的空间。不断地增删对象就会在文件中产生很多碎片，从而使得整个数据库文件的使用效率下降。压缩可以去除碎片，使 Access 2003 重新安排数据，收回空间。在对数据库文件压缩之前，系统会对文件进行错误检查，如果检测到数据库被损坏，就要求修复数据库。

数据库的压缩和修复可通过“工具”菜单中“数据库实用工具”命令中的“压缩和修复数据库”子命令来实现。

3. 拆分数据库

把完成的数据库应用系统共享给网络上的其他用户时，就会发现用户若想访问数据库中的数据，必须要把所需要的表、查询、报表、页和宏等数据库对象都复制到本地计算机中，这样很不方便。数据库拆分可以把数据库应用系统一分为二，将数据部分放在后端的

数据库服务器上，而前端的操作界面(如窗体和报表等)放在每一个想使用这个数据库的计算机上。这样用户可以在自己的机器上操作时，而数据库服务器负责传输数据，从而构成一个客户/服务器的应用模式。

在拆分数据库后，前端数据库窗口中表对象的名字前都有一个小箭头，说明这些表是连接到后端数据库的，这里的表只是一个空壳，里面没有任何数据。打开这些表时，Access 2003 会自动连接到后端的数据库上，取回数据。而在后端数据库中，只有数据表，其他数据库对象都放在前端数据库中。

4. 同步复制

同步复制是为数据库制作一个副本，这样可以使副本与原数据库保持同步更新。通过“工具”菜单中“同步复制”下的“创建副本”命令，可以创建数据库的副本，并将原数据库转变为设计母版。

当设计母版中数据库对象的结构发生改变后，通过“同步复制”命令下的“立即同步”子命令，可以使副本数据库保持同步更新。

5. MDE文件

如果数据库中包含 VBA 程序代码，在数据库保存为 MDE 文件时将编译所有模块、删除所有的源代码，并且压缩数据库，这有助于数据库应用系统的优化。

生成 MDE 文件可以确保数据库中的窗体、报表和 VBA 代码的安全性。在原始数据库中定义的数据库密码和用户级安全机制，在 MDE 文件中仍然有效。

把数据库保存为 MDE 文件后，如下的操作将不能进行：

- (1) 在设计窗口中查看、修改或创建窗体、报表或模块。
- (2) 增加、删除或更改数据库的引用。
- (3) 修改 VBA 程序的代码。
- (4) 使用“选项”对话框更改数据库的 VBA 项目的名称。
- (5) 导入或导出窗体、报表或模块。

注意，在生成 MDE 文件前，一定要保存原始的数据库文件。所有对窗体、报表、模块以及用户级别安全的修改都必须在原始数据库中进行，然后再生成 MDE 文件。

3.3 Access 2003 数据库安全性

Access 2003 提供了一些加强数据安全的保护措施。在本节中，将主要介绍如何运用这些措施来加强数据库的保护，例如，密码保护、安全机制和账户。

3.3.1 设置数据库密码

为数据库设置密码，防止非法用户擅自进入数据库。

1. 设置密码

设置数据库密码, 应考虑密码的内容, 不能使用易识别的信息作为密码, 最好设置多于六位的密码; 另外, 使用字母作密码时要考虑大小写, 比较好的做法是混合使用数字和字母, 例如 123x4w5y。

【例 3.1】 设置“学生”数据库的密码。

主要操作步骤如下:

(1) 关闭当前数据库。

不必退出 Access 2003 系统, 但要关闭要设置密码的数据库。

(2) 选择“文件”菜单下的“打开”命令。

(3) 在“打开”对话框中, 单击数据库文件, 或者在“文件名”文本框中输入数据库的名字, 单击“打开”按钮右侧下拉按钮, 弹出一个包括 4 种指定打开数据库方式的菜单, 如图 3.6 所示。

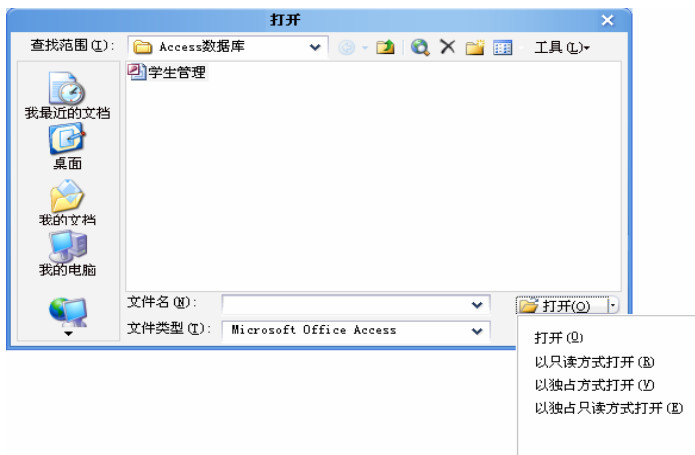


图 3.6 “打开”对话框

(4) 选择“以独占方式打开”命令, Access 2003 将按照指定的方式打开该数据库。

独占是对网络共享数据库数据的一种访问方式, 以独占模式打开数据库时, 将禁止他人打开该数据库的操作。

(5) 打开“工具”菜单, 选择“安全”命令中的“设置数据库密码”子命令, Access 2003 显示如图 3.7 所示的“设置数据库密码”对话框。

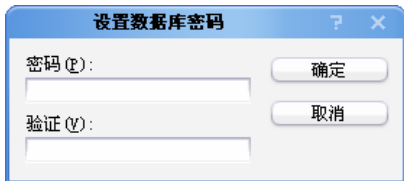


图 3.7 “设置数据库密码”对话框

(6) 在“密码”文本框中输入想要设置的密码; 在“验证”文本框再次输入相同的密码。

(7) 单击“确定”按钮。

这样就为数据库建立了打开密码，为了安全起见，在不使用数据库时，应关闭该数据库。



注意：

- 密码是与数据库一起保存的，将数据库复制或者移动到新位置，密码也随之移动。
- 要妥善保管数据库密码，如果忘记设定的密码，将不能再使用数据库。

2. 使用密码

当打开已经设置密码的数据库时，Access 2003 提示密码验证对话框，输入预先设定的密码，单击“确定”按钮即可打开数据库。

3. 撤销密码

如果需要撤销数据库的密码保护，可以撤销所设定的密码，具体操作步骤如下：

- (1) 关闭该数据库。
- (2) 选择“文件”菜单下的“打开”命令。
- (3) 在“打开”对话框中，单击数据库文件，或者在“文件名”文本框中输入数据库名字，单击“打开”按钮右侧下拉按钮，弹出一个包括 4 种指定打开数据库方式的菜单，选择“以独占方式打开”命令，输入预先设定的密码，打开数据库。
- (4) 打开“工具”菜单，选择“安全”命令中的“撤销数据库密码”子命令。
- (5) 在“密码”文本框中输入设定的密码，单击“确定”按钮。

3.3.2 用户级安全机制

Access 2003 用户级安全机制类似于服务器或主机系统的用户级安全机制，使用密码和权限，可以允许或限制个人、组对数据库中对象的访问操作。

安全账户定义了个人和组访问数据库中的对象的权限，这一信息存储在工作组信息文件中。在工作组信息文件中，通过用户的个人 ID 和密码将用户标识为已授权的单个用户，同时也标识为指定组的成员。用户启动 Access 2003 时要输入一个密码，系统将读取工作组信息文件。在该文件中每个用户都有唯一的代码标识，如果发现不是授权用户，则拒绝打开数据库；是授权用户，则可以根据其权限使用数据库。

1. 用户、组和权限

用户账户为个人提供特定的权限，以便访问数据库中的信息和资源。组是用户用于控制和管理这个组对数据库中对象的访问权限。一个组账户中包含若干用户账户，分配给一个组的权限适用于组中所有用户，定义不同的用户，并分配到不同的组中，为每个组分配相应的权限，可以达到限制不同权限用户对数据库实施不同级别操作的目的。

权限用于指定用户对数据库中的数据或对象所拥有的访问权限类型。

常用的权限类型有：打开/运行、以独占方式打开、读取对象的设计视图、使用对象的设计视图、管理员、读取数据、更新数据、插入数据和删除数据。

(1) 某些权限会自动隐含其他权限。例如对表的“更新数据”权限会自动隐含“读取数据”和“读取对象的设计视图”权限，因为只有具有这两项权限才能修改表中的数据。

“使用对象的设计视图”和“读取数据”权限则隐含了“读取对象的设计视图”权限。对宏的“读取对象的设计视图”权限则隐含了“打开/运行”权限。

(2) 在多用户环境中设计窗体、报表、宏和模块，必须以独占方式打开数据库，用户必须拥有对 Access 数据库的“以独占方式打开”的权限。

2. 工作组信息文件

(1) 工作组信息文件

存储有关工作组成员的信息，包括用户的账户名、密码及所属的组。Access 2003 在打开数据库时会读取工作组信息文件，以确定允许哪个用户访问数据库中的对象以及他们的权限。

第一次安装 Access 2003 数据库系统时，系统会自动生成一个默认的工作组信息文件。

可以修改默认工作组信息文件或为数据库创建新的工作组信息建立用户级安全机制。

(2) 信息文件中的账户

一个工作组信息文件中包含如下几个预定义的账户：

- ① 管理员：默认的用户账户。
- ② 管理员组：一个组账户。

账户中的所有成员都能够管理 Access 2003 数据库，管理员组中至少有一个管理员权限的账户。当最初建立数据库时，管理员组中只包含一个管理员账户 Administrator。

- ③ 用户组：一个组账户。

账户中的所有成员都可以使用 Access 2003 数据库。当使用管理员账户建立一个新用户账户时，新建账户将被自动加入到用户组中。

只有拥有数据库的管理员权限，才能在该数据库的任何工作组信息文件中加入另外的用户账户或组账户。

3.3.3 使用权限

账户的权限决定了该账户能够访问数据库中的哪些对象以及操作限制。

1. 显式权限和隐式权限

显式权限是直接授予用户账户的权限。

隐式权限是作为组成员继承组的权限。

2. 权限所适用对象

在 Access 2003 中使用不同的权限，要视对象而定，有一些权限适用于整个数据库，而另一些只适用于数据库中的单个对象。

例如,“打开/运行”权限,适用于数据库、窗体、报表等对象,而“读取数据”适用于表、查询等对象。

3.4 安全机制管理

3.4.1 增加账户

所创建的账户必须保存在这些用户要使用的工作组信息文件中,如果运行不同的工作组信息文件创建数据库,在创建账户前应更改工作组。可以使用“工作组管理员”(“工具”菜单中“安全”命令下)找出正在使用的工作组信息文件。

增加账户的主要操作步骤:

- (1) 打开数据库。
- (2) 选择“工具”菜单,再选择“安全”菜单项下的“用户与组账户”命令,打开“用户与组账户”对话框。
- (3) 选择“用户”选项卡,单击“新建”按钮,打开“新建用户/组”对话框。
- (4) 在“新建用户/组”对话框中,输入新账户名称和个人 ID(PID, PID 不是密码)然后,单击“确定”按钮,新账户创建完毕。
- (5) 从“可用的组”列表框中选择组,然后单击“添加”按钮,即可将当前用户(显示在“名称”框中)添加到组中。

3.4.2 删除账户

删除用户账户的操作也很简单,但管理员账户是不能删除的。

主要操作步骤:

- (1) 以管理员组成员的身份登录。
- (2) 打开数据库。
- (3) 选择“工具”菜单,再选择“安全”菜单项下的“用户与组账户”命令,打开“用户与组账户”对话框。
- (4) 在“用户”选项卡中,单击“名称”下拉列表框右侧下拉按钮,选择要删除的用户名,单击“删除”按钮。
- (5) 单击“是”按钮,单击“确定”按钮关闭对话框。

3.4.3 更改账户权限

如果要更改某个用户的权限,可以使用以下操作步骤:

- (1) 打开数据库。
- (2) 选择“工具”菜单,再选择“安全”菜单项下的“用户与组权限”命令,打开“用户与组权限”对话框。

(3) 在“用户名/组名”列表框中单击要修改权限的账户名,在“对象名称”列表框中单击要授权的对象,然后在“权限”选项组中更改权限。

(4) 单击“确定”按钮。

3.4.4 打印账户和组账户列表

可以打印用户账户和组账户用户列表,具体操作如下:

(1) 使用要打印的安全性信息的工作组信息文件启动 Access 2003 数据库。

(2) 通过使用“安全”菜单下的“工作组管理员”命令可以得知哪个工作组信息文件是当前的,或对工作组进行更改。

(3) 打开数据库。

(4) 选择“工具”菜单,选择“安全”菜单项下的“用户与组账户”命令,打开“用户与组账户”对话框。

(5) 在“用户”选项卡中,单击“打印用户和组”按钮。

(6) 在“打印安全性”对话框中,选择下列操作之一。

- 用户和组:打印用户账户和组账户的信息。
- 仅限于用户:打印当前工作组定义的所有用户。
- 仅限于组:打印当前工作组定义的所有组。

需说明的是:在建立、删除账户和更改账户权限时,一定要使用管理员账户进入数据库,只有管理员账户才有权限对其他账户进行增加、删除以及账户权限的设置。

3.5 加密数据库

数据库加密是指对数据库文件进行压缩,加密后的数据库无法通过工具程序或字处理程序解密。加密后的数据库依然可以在 Access 2003 系统中打开,不限制用户对对象的访问。

只有数据库的所有者,或者是具有“以独占方式打开”权限的管理员组的成员,才可以对数据库进行加密或解密,在数据库打开时不能对其进行加密或解密。数据库解密是加密的逆过程。

数据库加密的主要操作步骤:

(1) 启动 Access 2003,不打开数据库。

(2) 选择“工具”菜单,选择“安全”菜单项下的“编码/解码数据库”命令,打开“编码/解码数据库”对话框。

(3) 选择要编码/解码的数据库,单击“确定”按钮,打开“数据库编码后另存为”对话框。

(4) 输入加密或解密之后另存为的数据库名称及保存路径,单击“确定”按钮。

 **注意:** 如果在“数据库编码后另存为”对话框中使用原有的数据库名称和保存路径,在成功加密或解密后,Access 2003 会自动将原有的数据库替换为加密或解密后的版本,如果出现错误,Access 2003 将保留原有的数据库文件。

一般在加密数据库前，要做好原始数据库的备份，因为 Access 2003 会在加密全部成功后替换原始数据库。

习 题

1. Access 2003 数据库中包括哪 7 种对象？
2. 建立一个 Access 数据库的基本步骤是什么？
3. 在 Access 中，数据库文件的打开有哪 4 种形式？
4. 简述在 Access 中设置数据库的密码的步骤。
5. 简述建立完整的系统安全机制的操作步骤。
6. 简述在 Access 中加密数据库的步骤。

第4章 数据表设计与应用

本章学习目标

数据表(简称“表”)是数据记录的集合,是数据库最基本的组成部分。一个数据库里可以有多个数据表,它们包含了数据库的所有数据信息。一个表又由多个具有不同数据类型的字段组成。本章从数据表的创建入手,对表的管理与维护、表的操作和数据的导入与导出进行介绍。通过本章的学习,读者应掌握以下内容:

- 数据表的结构设计与字段属性设置
- 数据表的管理与维护
- 数据表的操作方法
- 数据的导入与导出

4.1 创建表

表是 Access 2003 数据库最基本的对象,其他的数据库对象,如查询、窗体和报表等都是在表的基础上建立并使用的。因此,表的结构是否合理,可以说是整个数据库的关键所在。一个数据库中可以建立多个表,通过在表之间建立关系,就可以将存储在不同表中的数据联系起来供用户使用。

Access 2003 提供三种创建表的方法:使用向导创建表、通过输入数据创建表和使用设计器创建表。这三种创建表的方法各有各的优点,适用于不同的场合。

4.1.1 数据表结构设计

数据表由表结构和表内容两部分组成,先建立表结构,然后才能输入数据。数据表结构设计主要包括:字段名称、字段类型和字段属性的设置。

设计表结构的主要工具是表设计器(又称为设计视图)。

1. 使用设计视图创建表

在 Access 2003 中,使用数据表设计视图,不仅可以创建表,而且可以修改已有表的结构。使用设计视图创建表的主要步骤与操作如下:

(1) 在数据库窗口的“对象”列表中单击“表”。

(2) 在右侧列表中双击“使用设计器创建表”;或者单击“新建”按钮,在出现的“新建表”对话框中选择“设计视图”选项,然后单击“确定”按钮,进入如图 4.1 所示的表设计视图。

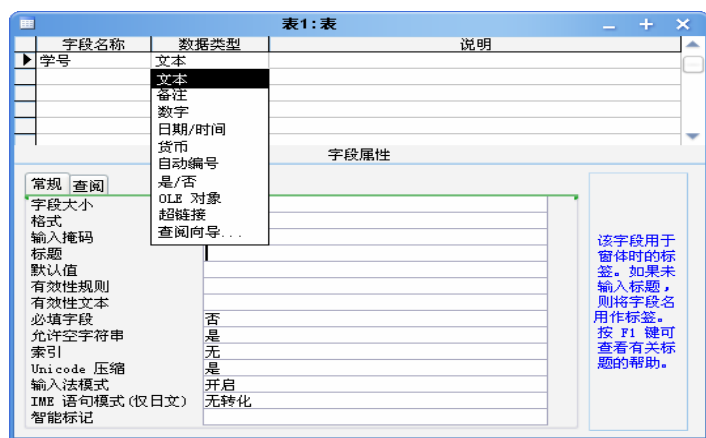


图 4.1 表设计视图

(3) 在表设计视图中，“字段名称”列输入字段名；“数据类型”列选择字段的数据类型；“说明”栏中输入有关此字段的说明；窗口下部的“字段属性”区用于设置字段的属性，例如，设置文本字段的“字段大小”来控制允许输入的最大字符数。每个字段可用属性取决于为该字段选择的数据类型。

(4) 当把数据表的所有字段名、数据类型、说明及字段属性等项都设置完毕后，可以选择“文件”菜单中的“保存”命令或单击工具栏上的“保存”按钮，在打开的“另存为”对话框中输入表的名称，然后单击“确定”按钮，完成数据表结构的设计。

(5) 如果在保存表之前未定义主码字段，则 Access 2003 将询问是否由系统自动添加一个主码，选择“是”表示确认，也可以选择“否”表示不需要自动添加主码字段。

2. 关于字段名称、数据类型、说明

(1) 字段名称

在“字段名称”列中可以添加或显示已有字段名称，字段名的命名必须符合 Access 2003 的对象命名规则。即：

- 长度最多可达 64 个字符。
- 可以包括字母、数字和空格，以及除句点、惊叹号、重音符号和方括号外的所有特殊字符。
- 不能使用前导空格或控制字符(ASCII 值从 0 至 31 的字符)。

(2) 数据类型

用于指定在字段中存储的数据类型。在 Access 2003 中，只允许选择使用系统提供的 10 种数据类型。

数据类型的种类与用途如表 4.1 所示。

表 4.1 数据类型及用途

数据类型	用 途
文本	字符和/或数字的任意组合，不能用于计算。最长 255 个字符，默认长度为 50 个字符

备注	超长的文本，用于注释或说明，最长 65535 个字符
----	----------------------------

续表

数据类型	用 途
数字	用于计算的值，1、2、4 或 8 字节(16 字节用于同步复制 ID)
日期 / 时间	表示日期和时间，可用于计算，最多 8 字节
货币	表示货币的数据类型，可用于计算，小数点左边最多为 15 位，右边可精确到 4 位，最多 8 字节
自动编号	Access 为每条记录提供唯一值的数值类型，常用做主关键字，4 字节(16 字节用于同步复制 ID)
是/否	布尔型，1 字节，如是/否、真/假、开/关等
OLE 对象	源于其他基于 Windows 应用程序的对象链接与嵌入，如 Excel 表单、Word 文档、图片、声音等，最多 1G 字节
超级链接	建立一个存储超级链接的字段，可以链接到一个 UNC 或 URL 字段，由 4 部分组成：显示文本、地址、子地址、屏幕提示，用#间隔。最多 2048 个字符
查询向导	创建一个字段，该字段允许从其他表、列表框或组合框中选择字段类型。输入表数据时，可从一个下拉列表中选择值

(3) 说明

说明列用于对字段进行必要的说明，例如，字段的含义及用途等。

3. 字段属性

字段属性决定了如何存储和显示字段中的数据。每种类型的字段都有一个特定的属性集。Access 2003 为每一个字段指定一些默认属性，可改变这些属性，字段的常规属性选项卡如表 4.2 所示。

表 4.2 “常规”属性选项卡

属性	作 用
字段大小	设置文本、数据和自动编号类型的字段中数据的范围，可设置的最大字符数为 255
格式	控制显示和打印数据格式，选择预定义格式或输入自定义格式
小数位数	指定数据的小数位数，默认值是“自动”，范围是 0~15
输入法模式	确定当焦点移至该字段时，准备设置的输入法模式
输入掩码	用于指导和规范用户输入数据的格式
标题	在各种视图中，可以通过对象的标题向用户提供帮助信息
默认值	指定数据的默认值，“自动编号”和“OLE”数据类型无此项属性
有效性规则	一个表达式，用户输入的数据必须满足该表达式
有效性文本	当输入的数据不符合有效性规则时，要显示的提示性信息

必填字段	该属性决定是否允许出现 Null 值
允许空字符串	决定“文本”和“备注”字段是否可以等于零长度字符串(“”)
索引	决定是否建立索引及索引的类型
Unicode 压缩	指定是否允许对该字段进行 Unicode 压缩

(1) 字段大小

字段大小用于设置存储字段中, 文本数据的最大长度或数值的取值范围, 只有文本和数值类型的字段可以选择。

文本类型的字段宽度范围为 1~255 个字符, 系统默认为 50 个字符。需注意的是: 对文本型字段, Access 2003 以实际输入的字符数来决定所需的磁盘存储空间。

数据类型的字段宽度如表 4.3 所示, 共有六种可选择的字段大小: 字节、整型、长整型、单精度型、双精度型和小数, 系统默认是长整型。

表 4.3 数据型字段大小的属性取值

类型	作用	小数位	占用空间
字节	0~255(无小数位)的数字		1 个字节
整型	-32768~32767(无小数位)的数字		2 个字节
长整型	-2147483648~2147483647(无小数位)的数字		4 个字节
单精度型	负值: $-3.4 \times 10^{38} \sim -1.4 \times 10^{-45}$ 的数字 正值: $1.4 \times 10^{-45} \sim 3.4 \times 10^{38}$ 的数字	7	4 个字节
双精度型	负值: $-1.8 \times 10^{308} \sim -4.9 \times 10^{-324}$ 的数字 正值: $1.8 \times 10^{-308} \sim 4.9 \times 10^{324}$ 的数字	15	8 个字节
小数	$-10^{28} - 1 \sim -10^{28} - 1$ 范围的数字	28	12 个字节

(2) 格式属性

格式属性用来规定文本、数字、日期和“是/否”型字段的数据显示或打印格式, 对存储数据不起作用, 也不检查无效输入。

对日期/时间型、数字型、货币型及“是/否”型, 系统提供了如表 4.4 所示的格式类型。

表 4.4 几种数据类型的字段格式设置

类型	格式类型与示例
日期/时间型	常规日期(默认设置): 1994-6-11 17: 40: 32 长日期: 1994 年 6 月 11 日 中日期: 94-06-11 短日期: 1994-6-11 长时间: 17: 40: 32 中时间: 下午 5: 40 短时间: 17: 40
数字/货币型	常规数字(默认值): 3456.789

	货币: ¥3,456.79 美元: \$3,456.79 固定: 3456.79 标准: 3,456.79 百分比: 123.00% 科学计数: 3.46E+03
--	--

续表

类型	格式类型与示例
是/否型	真/假: True(-1)/False(0) 是/否: Yes(-1)/No(0) 开/关: On(-1)/Of(0)
文本/备注型	@: 要求文本字符(字符或空格) &: 不要求文本字符 <: 使所有字符变为小写 >: 使所有字符变为大写

(3) 输入掩码

输入掩码用于指导和规范用户输入数据的格式,还可以控制文本框类型控件的输入值。如果为某个字段定义了输入掩码,同时又设置了格式属性,格式属性在数据显示时优先于输入掩码的设置。

输入掩码定义可包含用分号分隔的三个节,例如,定义日期型字段的输入掩码为 0000/99/99;0;" "。";" 用于区分节。掩码组成如表 4.5 所示。

表 4.5 掩码组成

节	含 义
第一个	输入掩码本身
第二个	决定是否保存原义显示字符。“0”表示以输入的值保存原义字符;“1 或空白”表示只保存输入的非空格字符
第三个	显示在输入掩码处的非空格字符,可以使用任何字符。“ ”代表一个空格。如果省略该节,将显示下划线(_)

输入掩码由用来分隔输入空格的原义字符(例如空格、点、点划线、和括号)组成。“输入掩码”属性设置则由文本字符和特殊字符组成,特殊字符将决定输入的数值类型。输入掩码主要用于“文本”和“日期/时间”型字段,但也可以用于“数字”或“货币”字段。

Access 2003 将按表 4.6 所示转译“输入掩码”属性定义的第一个节中的字符。如果要定义原义字符,可输入该表以外的任何其他字符,包括空格和标号。如果要将字符中的某一个定义为原义字符,可在字符前面加上“\”。

例如,对于“入校日期”字段,为方便用户输入,可以预留年、月、日输入区,并用

输入掩码定义字符分隔年、月、日。如设置其输入掩码为：0000/99/99;0;#。第一节“0000/99/99”中，“0”表示此处只能输入一个数，而且必须输入；“9”代表此处只能输入一个数，但不是必须输入；“/”符号为转意分隔符，输入数据时直接跳过。第三节中的“=”为占位符。

输入掩码属性设置可以使用向导或在输入掩码属性框中直接输入，但系统仅对“文本”和“日期/事件”型字段提供向导，其他数据类型没有向导帮助。

表 4.6 输入掩码定义字符

字符	说 明
0	必须输入数字(0 ~ 9)
9	可以输入数字或空格
#	可以选择输入数字或空格(空白将转换为空格，允许使用加号和减号)
L	必须输入字母(A ~ Z)
?	可以选择输入字母(A~ Z)
A	必须输入字母或数字
a	可以选择输入字母或数字
&	必须输入任一字符或空格
C	可以选择输入任一字符或空格
. , : ; - /	十进制占位符和千位、日期与时间分隔符(实际使用的字符取决于Windows“控制面板”的“区域设置属性”中指定的区域设置)
<	使其后所有的字符转换为小写
>	使其后所有的字符转换为大写
\	使其后的字符显示为原义字符。可用于将该表中的任何字符显示为原义字符(例如，\A 显示为 A)
!	使输入掩码从右到左显示，而不是从左到右显示。输入掩码中的字符始终都是从左到右显示
密码	将“输入掩码”属性设置为“密码”，以创建密码项文本框。文本框中输入的任何字符都按字面字符保存，但显示为星号(*)

对于“数字”或“货币”等类型的字段，只能使用字符直接定义输入掩码属性。例如，要定义“课程”表中的“学分”字段的输入掩码为 9，可以在“课程”表的设计视图中，选择“学分”字段，在其输入掩码文本框中直接输入 9。此设置意味着不管“学分”字段定义的宽度是多少位，在输入数据时仅接受 0 到 9 之间的数字，且只能输入一位。

当然，对于“文本”和“日期/事件”型字段也可以不用向导而使用字符直接定义。例如，在“入学日期”字段的输入掩码文本框中直接输入 0000/99/99。

使用“输入掩码向导”来完成掩码的输入，设置方法比较简单，在此不再详细说明。

(4) 标题

设置“标题”属性值，在显示表数据时，表列的栏目名称将显示“标题”属性值，而不显示字段名称。

(5) 默认值

为字段设置默认值，在向表中增加记录时，Access 2003 自动为字段填入设定的默认值。

(6) 有效性规则与有效性文本

有效性规则是指一个表达式，用户输入的数据必须满足表达式，使表达式的值为真，当焦点离开此字段时，Access 2003 会检测输入的数据是否满足有效性规则，如不满足，则根据“有效性文本”设置的内容提示相应信息。

有效性规则表达式包括一个运算符和一个比较值，当运算符为“=”时，可省略不写。常用的运算符如表 4.7 所示。

表 4.7 在有效性规则中使用的运算符

运算符	意 义
<	小于
<=	小于等于
>	大于
>=	大于等于
=	等于
<>	不等于
In	所输入数据必须等于列表中的任意成员
Between	“Between A and B”代表所输入的值必须在 A 和 B 之间
Like	必须符合与之匹配的标准文本样式

有效性文本的设定内容是当输入值不满足有效性规则时，系统提示的信息。有效性规则和有效性文本通常是结合起来使用的。

【例 4.1】 如下示例给出了关于有效性规则 and 有效性文本的设置。

有效性规则设置	有效性文本设置
<>0	请输入一个非零值
0 or >=60	值必须为 0 或大于等于 600
<#1/1/2005#	输入一个 2005 年之前的日期
>=#1/1/2004# and <#1/1/2005#	日期必须是在 2004 年内
Like "C???"	值必须是以 C 打头的四个字符

(7) 必填字段

“必填字段”属性取值只有“是”和“否”两项。当设置为“是”时，表示必须在字段中输入内容，不允许本字段为空。

(8) 允许空字符串

该属性仅对“文本”型字段有效，取值只有“是”和“否”两项，当设置为“是”时，表示字段可以不填写任何字符。

(9) 索引

用于设置单一字段索引。索引用于提高对索引字段的查询速度及加快排序与分组操作。

共有如下三项取值：

“无”：表示本字段无索引。

“有(有重复)”：表示本字段有索引，但允许表中该字段数据重复。

“有(无重复)”：表示本字段有索引，但不允许表中该字段数据重复。

(10) Unicode 压缩

取值只有“是”和“否”两项，当设置为“是”时，表示本字段中的数据可以存储和显示多种语言的文本。

(11) 输入法模式

常用“开启”和“关闭”选项，若选择“开启”，则在向表中输入数据时，一旦该字段获得焦点，将自动打开设定的输入法。

4. 查阅属性

字段的查阅属性选项卡只有一个“显示控件”属性，该属性仅对文本、数字和“是/否”类型的字段有效。

对文本和数字类型的字段提供了 3 个选项值：文本框(默认值)、列表框和组合框；为“是/否”类型的字段提供了 3 个选项值：复选框(默认值)、文本框和组合框，如图 4.2 所示。

图 4.2 字段查阅属性参数

若对文本和数字类型的字段使用列表框或组合框，以及对“是/否”类型字段使用组合框，意味着可以与其他表或查询结合来向字段中输入数据。

若“是/否”类型字段的显示控件属性选择复选框，在输入字段值时，以“✓”显示代表“真”值，以“□”显示代表“假”值。此时，“常规”选项卡中格式属性的设置不起作用。

若“是/否”类型字段的显示控件属性选择文本框，“常规”选项卡中格式属性没有设置，则字段以“-1”代表“真”值，以“0”代表“假”值，如果“常规”选项卡中格式属性已设置，其设定值将代表真假值的文字显示。

需说明的是：使用字段的“查阅向导”数据类型或设置字段的“字段属性”与“查阅”选项卡，对一些具有可选择输入数据的字段，例如“系别”、“专业”和“政治面貌”等，会给数据输入带来很大方便，特别是应用系统的初始数据录入，不必编写程序，即可提供方便的数据录入操作。

4.1.2 主码

主码(也称主键)是用于唯一标识表中每条记录的一个或一组字段。Access 2003 建议每一个表设计一个主码，这样在执行查询时用主码作为主索引可以加快查找的速度。还可以利用主码定义多个表之间的关系，以便检索存储在不同表中的数据。

表设计了主码，可以确保唯一性，即避免任何重复的数值或 Null(空)值保存到主码字段中。在 Access 2003 中，可以定义三种主码：自动编号、单字段及多字段。

1. 自动编号主码

创建一个空表时，在保存表之前如果未设置表的主码，Access 2003 会询问是否需要设置一个自动编号的主码。它的作用是在表中添加一个自动编号字段，在输入记录时，自动编号字段可设置为自动输入连续数字的编号。

2. 单字段主码

在表中，如果某一字段的值能唯一标识一条记录，就可以将此字段指定为主码。如果选择作为主码的字段有重复值或 Null(空)值，Access 2003 就不会将它设置为主码。

3. 多字段主码

在表中，可以将两个或更多的字段指定为主码(至多包括 10 个字段)。例如，在学生“成绩”表中，“学号”与“课程号”字段的值可能都不是唯一的，因为一个学生可以选多门课，而一门课可以被多个学生选择。如果“学号”与“课程号”两个字段组成的字段组合指定为主码，就有唯一的值，并成为每一条记录的标识。

设置主码的方法非常简单：

(1) 在设计视图中打开表。

(2) 单击行选定器，选择主码字段所在的行，如果要设置多字段主码，先按下 Ctrl 键，然后单击行选定器选择所需的字段。

(3) 单击工具栏中的“主码”按钮或使用右击快捷菜单，选择“主码”命令。主码指示符将出现在该行的字段选择器上，表明已经将该字段设置为主码。

4.1.3 索引

使用索引就如同使用一本书的目录，可以在表中快速查找所需的数据。Access 2003 允许用户基于单个字段或多个字段创建记录的索引，一般可以将经常用于搜索或排序的单个字段设置为单字段索引；如果要同时搜索或排序两个或两个以上的字段，可以创建多字

段索引，多字段索引能够区分与第一个字段值相同的记录。

1. 创建索引

Access 2003 将表中的主码自动创建为索引。如果要创建某个字段或字段的组合为索引，可按下述步骤进行：

(1) 创建单字段索引

- ① 在设计视图中，打开需要设置单字段索引的表。
- ② 单击要设置索引的字段行。
- ③ 在“字段属性”选项卡中，单击“索引”属性框，根据字段的数据值，选择“有(无重复)”或“有(有重复)”选项。

(2) 创建多字段索引

使用多字段索引排序记录时，Access 2003 将首先使用定义在索引中的第一个字段进行排序，如果记录在第一个字段中的值相同，使用索引中的第二个字段进行排序，依次类推。

- ① 在设计视图中，打开需要创建多字段索引的表。
- ② 选择“视图”菜单中的“索引”命令，或者单击工具栏中的“索引”按钮，打开如图 4.3 所示的“索引”窗口。如果当前表已定义了主码，Access 自动在“索引”窗口的第一行显示出主码索引的名称、字段名称以及排序次序。
- ③ 在“索引名称”列中输入索引名称。
- ④ 在“字段名称”列中，单击右边向下箭头，从下拉列表中选择索引的第一字段。
- ⑤ 在“排序次序”列中，选择“升序”或“降序”选项。
- ⑥ 根据需要可继续定义其他需要索引的字段。



图 4.3 “索引”窗口

2. 查看与编辑索引

如果要查看或编辑表中已有的索引，可以按照下述步骤进行：

- (1) 在设计视图中打开包含索引的表。
- (2) 选择“视图”菜单中的“索引”命令，或单击工具栏中的“索引”按钮，打开“索

引”窗口。

(3) 在“索引”窗口中列出当前表中已定义的索引名称、索引字段、索引的排序次序以及索引属性等设置，用户可以根据需要更改这些设置。

(4) 要删除某个索引，可以单击行选定器选择索引，然后按 Delete 键删除。

4.1.4 使用向导创建表

在 Access 2003 的创建表向导中，有各种各样预定义的示例表，如客户表、雇员表、产品表、订单表等，用户可以利用这些示例表为模板创建新表。

使用向导创建表的操作步骤如下：

(1) 在数据库窗口中选择“表”对象。

(2) 直接双击“使用向导创建表”或单击“新建”按钮，打开如图 4.4 所示的“新建表”对话框，选择“表向导”后单击“确定”按钮。

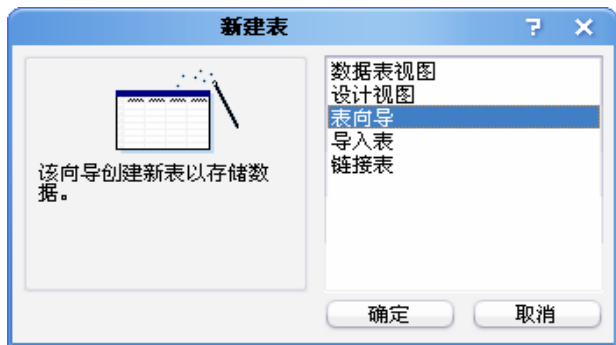


图 4.4 “新建表”对话框

(3) 按向导提示进行操作，逐步完成表结构的设计，由于向导操作提示详细，在此不给出每步的操作说明与界面。

4.1.5 通过输入数据创建表

Access 2003 允许用户先不创建表，而是通过输入一组数据，由系统根据输入数据的特点自动确定各个字段的数据类型和长度，从而创建一个新表。若对表结构的设计不满意，可以在设计视图中进行修改。

通过输入数据创建表的操作步骤如下：

- (1) 在数据库窗口中选择“表”对象。
- (2) 双击“通过输入数据创建表”，系统自动打开一个空表，如图 4.5 所示。系统默认为 10 字段，可以插入或删除字段，各个字段的默认名称依次是字段 1、字段 2，……，字段 10。



图 4.5 输入数据创建表

- (3) 如果要对字段重新命名，可双击字段名，然后输入新名称。
- (4) 在各字段中输入相应的数据。输入完数据后，单击工具栏中的“保存”按钮，或者选择“文件”菜单中的“保存”命令，在显示的“另存为”对话框中输入表的名称，单击“确定”按钮保存表。

4.2 管理与维护表

4.2.1 表间关系的建立

查询数据库数据时，经常要在两个或多个表的字段中查找和显示数据记录。表间的记录联接靠建立表间关系来保证，所以，指定表间的关系是非常重要的。一般情况下，如果两个表使用了共同的字段，就应该为这两个表建立一个关系，通过表间关系就可以指出一个表中的数据与另一个表中的数据的相关方式。

表间关系的类型有：一对一、一对多和多对多三种。

当创建表间关系时，必须遵从“参照完整性”规则，这是一组控制删除或修改相关表数据方式的规则。

参照完整性规则：

- (1) 在将记录添加到相关表中之前，主表中必须已经存在了匹配的记录。
- (2) 如果匹配的记录存在于相关表中，则不能更改主表中的主码值。
- (3) 如果匹配记录存在于相关表中，则不能删除主表中的记录。

1. 创建表间关系

具体操作步骤如下：

(1) 在数据库窗口中，单击工具栏上的“关系”按钮，或在数据库窗口中右击，在弹出的快捷菜单中选择“关系”命令，系统将打开如图 4.6 所示的“关系”窗口。如果在数据库中已经创建了关系，关系窗口中将显示出这些关系；如果数据库中还没有定义任何关系，Access 2003 会打开一个空白关系窗口。

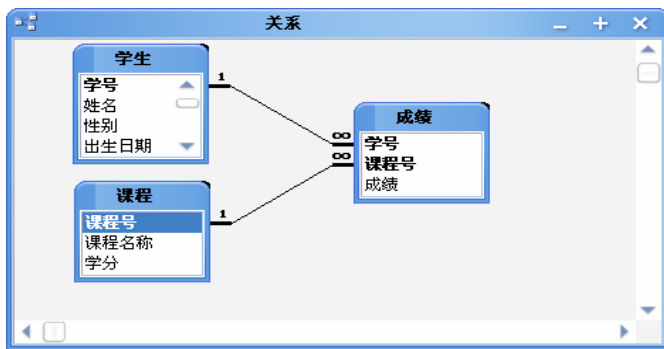


图 4.6 学生管理数据库的“关系”窗口

(2) 在窗口中右击并在弹出的菜单中选择“显示表”命令，打开如图 4.7 所示“显示表”对话框。

“显示表”对话框有三个选项卡：

- “表”选项卡：列出数据库中所有的数据表。
- “查询”选项卡：列出数据库中所有的查询。

- “两者都有”选项卡：列出数据库中所有的数据表和查询。

可以从中选择需要创建关系的表或查询，把它们添加到关系窗口中。例如，分别添加学生、课程和成绩表，此时，关系窗口呈现如图 4.8 所示的布局。

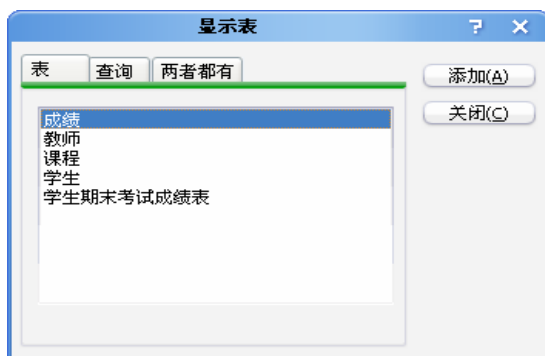


图 4.7 “显示表”对话框



图 4.8 添加表后的关系窗口

(3) 在关系窗口中，将显示添加表的字段，主码字段名用粗体标识。一般在主码和外码之间建立表间关系，即被参照关系的主码与参照关系的外码之间，两者一般设计为字段名相同。

例如，建立学生与成绩表的关系。单击学生表的“学号”字段且按住不放，然后把它拖到成绩表中“学号”字段上，当释放鼠标时，系统打开“编辑关系”对话框，如图 4.9 所示。

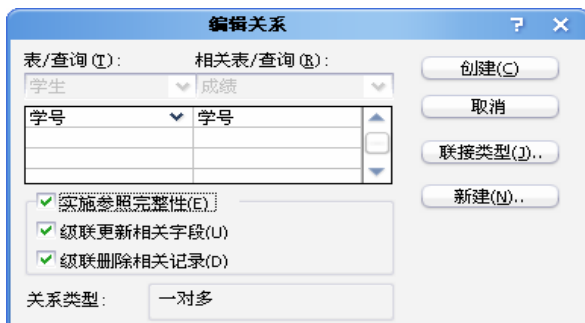


图 4.9 “编辑关系”对话框

(4) 在“编辑关系”对话框中选中“实施参照完整性”和“级联更新相关字段”复选框，即当更新主表中关键字段的内容时，同步更新关系表中相关内容。若选择“级联删除相关记录”，当在删除主表中某个记录时，同步删除关系表中相关记录。

(5) 单击“联接类型”按钮，打开“联接属性”对话框，如图 4.10 所示，在此选择联接的方式。用户可以根据实际需要进行选择，系统默认选择为第一种，在此选择系统默认设置。

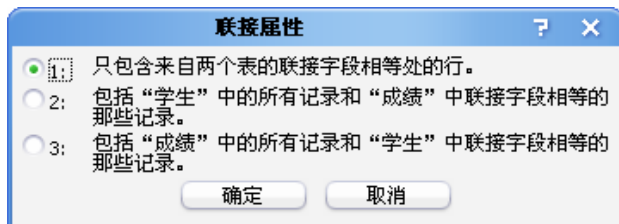


图 4.10 设置“联接属性”

(6) 在“编辑关系”对话框中，单击“创建”按钮，即在关系表之间有一条线将二者连接起来，表示已创建好表之间的关系。

(7) 关闭“编辑关系”对话框，保存设定的关系。

2. 编辑与删除表间关联

对已存在的关系，单击关系连线，连线会变黑，右击并选择“编辑关系”或双击，系统会打开“编辑关系”对话框，用户可以对创建的关系进行修改。

单击关系连线后按 Delete 键或右击并选择“删除”命令，可删除表间的关联。

4.2.2 修改表结构

在使用表之前，应该认真考察表的结构，查看表的设计是否合理，然后才能向表中输入数据或者基于表创建其他的数据库对象。

1. 添加、删除、重命名字段和移动字段的位置

在“设计”视图中，可以很容易添加、删除或重命名字段。

(1) 添加字段

在表的“设计”视图中，可以进行添加新字段的操作。

- 单击最后一个字段下面的行，然后在字段列表的底端输入新的字段名。
- 如果要在某一字段之前添加字段，单击要插入新行的位置。选择“插入”菜单中的“行”命令，或者右击并从弹出的快捷菜单中选择“插入行”命令，即可在当前字段之前出现一个空行，输入新的字段名并设置相应属性。

(2) 删除字段

如果要删除某一字段，单击该字段名，然后选择“编辑”菜单中的“删除行”命令，即可删除该字段，同时也删除该字段中的数据。

(3) 重命名字段

改变表中字段的名称并不会影响该字段的数据，但是会直接影响到其他基于该表创建的数据库对象，其他数据库对象对表中该字段的引用必须做相应的修改后方可生效。

在“设计”视图中，单击要重命名的字段，输入新的字段名称，然后单击工具栏中的“保存”按钮。

(4) 移动字段的位置

要改变字段的顺序，可单击行选择器以选中该行，然后拖动行选择器把字段移动到新位置。

 注意：可同时移动连续的多行，但不能同时移动非连续的多个字段。

2. 修改字段的数据类型

修改字段的数据类型将会造成表中数据的丢失。因此，在对包含数据的表进行数据类型的修改之前，应先做好表的备份工作。

在“设计”视图中，可方便地修改某字段的数据类型。

为了使所做的修改生效，应单击工具栏中的“保存”按钮，系统打开一个对话框。如果单击“是”按钮，不能进行转换的现有数据被从该字段中清除。如果单击“否”按钮，则将该数据恢复为原来的类型。

3. 修改字段的属性

字段属性是一个字段的特征集合，它们控制着字段如何工作。

在“设计”视图中，通过字段属性的“常规”与“查阅”选项卡，可以修改或重新设置字段的各项属性。

4.3 操作表

4.3.1 数据输入

在数据库窗口中，选中要打开的数据表，在表名称上双击，或单击工具栏上的“打开”按钮，即进入数据表视图窗口。窗口的主要组件说明：

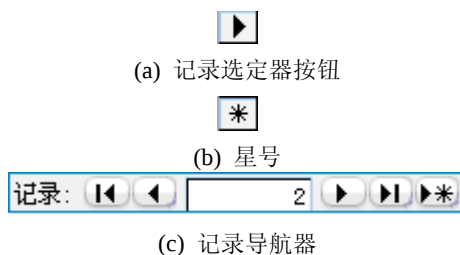


图 4.11 数据表视图窗口的主要组件

记录选定器按钮如(图 4.11(a))所示：数据表视图最左边的一列灰色按钮，用于选定记录。

星号(如图 4.11(b)所示)：出现在数据表视图最后一个记录的选择按钮上，用来表示这是一个假设追加记录。以只读方式打开数据库，在数据表视图中不出现假设追加记录。

记录导航器如图 4.11(c)所示：在数据表视图的底端，用于导航记录。

由于文本、数字和货币型字段数据输入比较

简单，在此不特别给出说明。

1. 输入“是/否”型数据

对“是/否”型字段，输入数据时显示一个复选框。选中表示输入“是(-1)”，不选中表示输入了“否(0)”，例如，“性别”字段。

但为了使显示内容更明确，即“性别”字段显示“男”或“女”，而真正存储到字段中的数据仍是“-1”或“0”，可以使用“查阅向导”数据类型来实现这种功能。

2. 输入“日期/时间”型数据

输入“日期/时间”型数据，不需要将整个日期全部输入，系统会按输入掩码来规范输入格式；按格式属性中的定义显示数据。

例如，在出生年月字段中输入“95-12-5”，若格式属性设置“长日期”，则会自动显示为“1995年12月5日”。

3. 输入“OLE对象”型数据

OLE 对象类型的字段使用插入对象的方式输入数据。

例如，学生表中的“照片”字段。当光标位于该字段时，右击并选择快捷菜单中的“插入对象”命令，打开“插入对象”对话框，如图 4.12 所示。



图 4.12 插入对象

选择“新建”单选按钮，则对话框会显示各种对象类型，可以选择某种类型创建新的对象，并插入到字段中。

选择“由文件创建”单选按钮，可以选择一个已存储的图片文件插入到字段中。

4. 输入“超链接”型数据

可以使用“插入超链接”对话框，实现超链接型字段的数据输入。

当光标位于该字段时，右击并选择“超链接”下的“编辑超链接”命令，打开“插入超链接”对话框，如图 4.13 所示。

在对话框中可以选择三种超链接：原有文件或网页、新建页和电子邮件地址。根据实际需要，选择输入“超链接”型字段的数据。



图 4.13 “插入超链接”对话框

5. 输入“查阅向导”型数据

如果字段的内容取自一组固定的数据，可以使用“查阅向导”数据类型。

例如，学生表中“性别”字段，可以选择为“是/否”型，但在输入数据时，要使用意义不明确的复选框。如选择“性别”字段为“查阅向导”数据类型，输入数据时，可以从组合框中选择“男”或“女”，而实际存储的可以设置为0或-1。

【例 4.2】 设计“性别”字段为“查阅向导”数据类型。

具体设计方法如下：

(1) 在学生表设计视图中，选择“性别”字段的数据类型为“查阅向导”，系统弹出“查阅向导”对话框，选择“自行键入所需的值”，打开如图 4.14 所示对话框。

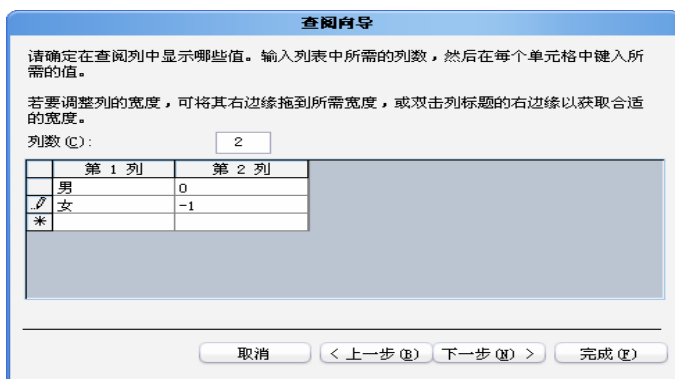


图 4.14 “查阅向导”对话框

在列数文本框中输入2，第1列中输入男、女，第2列中输入0、-1。

(2) 在组合框列表中，若希望只显示第1列中的男、女，而不显示包含实际数据的第2列，可将第2列的宽度调整为0。

(3) 选择哪一列要保存到数据表中，这里选择第2列，即“性别”字段保存的实际值为0、-1。

(4) 在查阅向导的“请为查阅列指定标签：”文本框中，保留“性别”，单击“完成”按钮，完成“性别”字段的查阅属性设置。

创建了“性别”字段的值列表后，在数据表视图中，“性别”字段值的输入方式为组合框式的选择。

4.3.2 排序记录

排序就是按照某个字段的内容值重新排列数据记录。默认情况下，Access 2003 按主码字段排序记录，如果表中没有主码，则以输入的次序排序记录。在数据检索和显示时，可按不同的顺序来排列记录。

1. 单字段排序

若要对表或窗体中的单个字段排序，先单击要排序的字段，然后进行下列操作之一：

- 单击工具栏中的“升序”或“降序”按钮。
- 使用“记录”菜单中的“排序”命令，选择“升序排序”或“降序排序”。
- 右击字段并从快捷菜单中选择“升序排序”或“降序排序”。

若要将记录恢复到原来的顺序，使用“记录”菜单中的“取消筛选/排序”命令，或右击数据表并从快捷菜单中选择“取消筛选/排序”。

2. 多字段排序

如果要将两个以上的字段排序，这些字段在数据表中必须相邻。排序的优先权从左到右。在确保要排序的字段相邻后，选择这些字段，再选择“升序”或“降序”排列。



注意： 对多字段排序时，排序字段按相同的顺序(升序或降序)排序，不会把升序和降序混合起来。要把排序顺序混合起来，应使用后面介绍的“高级筛选/排序”操作。

3. 保存排序顺序

改变记录的排序后，在关闭表时，Access 2003 会提示是否保存对设计(包括排序顺序)的更改，回答“是”，就保存排序，再打开时将按该排序显示。

4.3.3 筛选记录

当要显示数据表或窗体中的某些而不是全部记录时，可使用筛选操作。筛选处理是对记录进行筛选，选择符合准则的记录，准则是一个条件集，用来限制某个记录子集的显示。Access 2003 提供了五种筛选记录的方法：

- “按窗体筛选”：按输入到表框架的准则筛选记录。
- “按选定内容筛选”：显示与所选记录字段中的值相同的记录。
- “内容排除筛选”：显示与所选记录字段中的值不相同的记录。
- “高级筛选/排序”：除筛选外，可规定一个复合排序，以不同的顺序(升序或降序)对两个或多个字段排序。
- “输入筛选”：显示快捷菜单输入框，直接输入筛选准则。

1. 按选定内容筛选

按选定内容筛选是应用筛选中最简单和快速的方法，可以选择数据表的部分数据建立筛选准则，Access 2003 将只显示与所选数据匹配的记录。

在数据表视图中，打开数据表，选择记录中要参加筛选的一个字段中的全部或部分内容，使用“记录”菜单中的“筛选”操作，选择“按内容筛选”命令即可。例如，在“学生”表中，可以选择民族字段中的“回族”，应用筛选后，在数据表视图中仅显示民族为“回族”的学生记录。如果选择“内容排除筛选”，则显示学生表中民族字段不含“回族”的所有记录。

2. 按窗体筛选

可以在表的一个空白窗体中输入筛选准则，显示表中与准则相匹配的记录。打开数据表视图，单击工具栏上的“按窗体筛选”按钮，或使用“记录”菜单中的“按窗体筛选”命令，系统弹出按窗体筛选窗口，如图 4.15 所示。

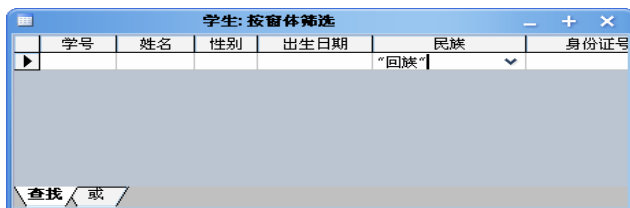


图 4.15 “按窗体筛选”窗口

从字段列表中选择要搜索的一个或多个字段的值，然后选择“筛选”菜单中的“应用筛选/排序”命令，系统会自动执行所设定的筛选，并显示筛选结果。使用“记录”菜单中的“取消筛选/排序”命令，可以取消本次筛选操作。

3. 高级筛选/排序

使用“高级筛选/排序”操作，可以对一个或多个数据表、查询进行筛选，还可以在一个或多个字段上添加排序次序。

打开数据表视图，使用“记录”菜单中的“筛选”操作，选择“高级筛选/排序”命令，打开“高级筛选/排序”操作窗口，如图 4.16 所示。

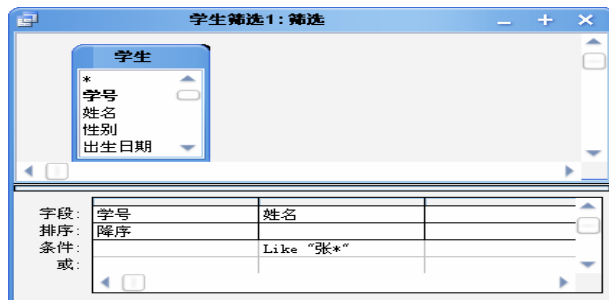


图 4.16 “高级筛选/排序”窗口

在“字段”栏中选择用于筛选的字段，在“条件”栏输入想要筛选的数值或表达式，单击工具栏或菜单中的“应用筛选/排序”命令，筛选结果将显示在数据表视图中。

4.3.4 设置数据表格式

1. 设置行高和列宽

(1) 设定行高参数

将鼠标指针停留在表中任一行，选择“格式”菜单中的“行高”命令，在弹出的“行高”对话框中，设定行高参数，为数据表指定行高。

(2) 手动调节列宽

将鼠标移至表中两个字段名的交界处，按住鼠标左键不放左右拖曳，可改变表的列宽。

(3) 设定列宽参数

将鼠标移至表中需要更改列宽的那一列中，使用“格式”菜单中的“列宽”命令，在弹出的“列宽”对话框中，设定行高参数或选择标准宽度，为选择的列设定列宽。

2. 数据字体的设置

数据表视图中的所有字体(包括字段数据和字段名)，其默认值均为宋体、常规、小五号字、黑色、无下划线。若需要更改字体设置，使用“格式”菜单中的“字体”命令，可以设定数据表视图的数据显示字体。

3. 表格样式的设置

数据表视图的默认表格样式为白底、黑字、银白色表格线构成的具有平面单元格效果的数据表形式。使用“格式”菜单中的“数据表”命令，可以设置数据表格式。

4. 隐藏列

(1) 设置列宽为零

将那些需要隐藏的字段列宽度设置为 0，这些字段列就成为隐藏列。

(2) 设置隐藏列

选择“格式”菜单栏中的“隐藏列”命令，可以很方便地将当前光标所在列隐藏起来。选择“取消隐藏列”命令，然后指定需要取消的隐藏列，可使得已经隐藏的列恢复原来设置的宽度。

5. 冻结列

如果数据表字段很多，有些字段只有通过滚动条才能看到。若想总能看到某些列，可以将其冻结，使在滚动字段时，这些列在屏幕上固定不动。

冻结列的操作方法是，在第一个需冻结的列的字段名上拖曳鼠标至最后一个需冻结的列的字段名上，即选定了所有需要冻结的列。选择“格式”菜单中“冻结列”命令，就完成了冻结列的操作。

选择“格式”菜单中“取消对所有列的冻结”命令，可以取消对所有列的冻结。

4.4 数据的导入与导出

使用数据的导入、导出和链接功能,可以将外部数据源如 Access 数据库、文本文件、Excel、FoxPro、ODBC、SQL Server 数据库等的数据库,直接添加到当前的 Access 数据库中,或者将 Access 数据库中的对象复制到其他格式的数据文件中。

4.4.1 导入、导出数据

1. 导入数据

使用导入操作可以将外部数据源数据变为 Access 格式。

导入操作的主要步骤如下:

- (1) 打开自己的数据库,即目标数据库。
- (2) 选择“文件”菜单“获取外部数据”下的“导入”命令,在打开的导入对话框中,选择需要导入的文件类型及文件名。例如,选择 Excel 电子表格文件“学生成绩单”。
- (3) 单击“导入”按钮,系统打开如图 4.17 所示的“导入数据表向导”对话框。

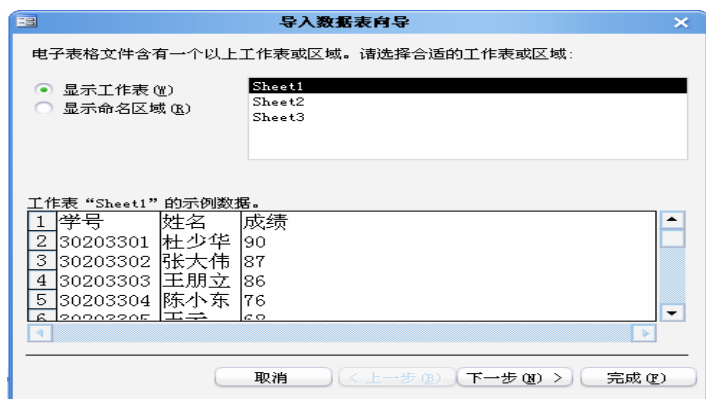


图 4.17 “导入数据表向导”对话框

(4) 单击“下一步”按钮,确认是否采用 Excel 表的第一行(列标题)作为数据表的字段名。

(5) 单击“下一步”按钮,选择“新表中”,将创建一个新的数据表;选择“现有的表中”,将导入的数据放入一个现有数据表中。

(6) 若选择“新表中”,在随后打开的对话框中,可对字段名、数据类型等进行相应修改及设置主码等,最后为导入的新表命名,单击“完成”按钮,便会在数据库窗口中添加一个新表。

由于导入外部数据的类型不同,导入的操作步骤也会有所不同,但基本方法是一致的。

2. 导出数据

导出数据是将 Access 数据库中的表、查询或报表复制到其他格式的数据文件中。

在数据库窗口中，选定某个数据表，选择“文件”菜单中的“导出”命令，在打开的对话框中选择文件的类型，文件的存储位置及文件名称即可。

4.4.2 链接数据

链接数据就是在源数据和目标数据之间建立一个同步的影像，所有对外部数据源数据的改动都会及时地反映到目标数据库中，同时，如果在 Access 中对链接数据进行了修改也会同步反映到外部数据源数据中。

链接数据操作的主要步骤：先打开自己的数据库，选择“文件”菜单“获取外部数据”下的“链接表”命令，在打开的对话框中，选择需要链接的文件及相关选项。这样便可在当前数据库中建立一个与外部数据链接的表。

若要取消链接，只需在数据库窗口中删除链接表即可。删除链接表并不影响外部数据表本身。

4.5 数据库的转换

Microsoft Access 2003 数据库管理系统允许在 Access 2003 数据库与旧版本数据库之间互相转换，用户可以打开使用旧版本 Access 创建的数据库，并且能够查看数据库对象，添加、删除或修改记录等，但是不能修改该数据库中各个对象的设计。如果用户要在 Access 2003 中修改已有对象的设计，可以将旧版本的数据库转换为 Access 2003 格式的数据库。

4.5.1 在 Access 2003 中使用旧版本的数据库

在 Access 2003 中可以使用 Access 95、Access 97 以及 Access 2000 等旧版本的数据库，但使用旧版本的 Access 并不能打开 Access 2003 创建的数据库。

在 Access 2003 中使用旧版本数据库的具体操作步骤如下：

(1) 选择“文件”菜单中的“打开”命令，或者单击“数据库”工具栏中的“打开”按钮，打开“打开”对话框。

(2) 在“打开”对话框中指定先前版本数据库的路径及文件名。单击“打开”按钮，将出现如图 4.18 所示的“转换/打开数据库”对话框。

(3) 在此对话框中，用户可以选择 Access 提供的两个选项。

- “转换数据库”选项：将旧版本的数据库转换为 Access 2003 数据库。
- “打开数据库”选项：在 Access 2003 中打开并查看选定的旧版本数据库，但不能修改该数据库中的任何数据库对象的设计。

(4) 选择“打开数据库”后单击“确定”按钮，即可在 Access 2003 中使用旧版本的数据库。

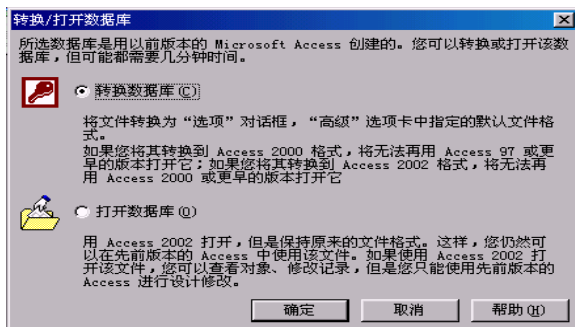


图 4.18 “转换/打开数据库”对话框

在 Access 2003 中使用旧版本的数据库后，用户再次打开该数据库时，系统不会再出现“转换/打开数据库”对话框，而是直接弹出提示不能修改数据库对象的对话框。

4.5.2 将Access 2003 数据库转换为Access 97 或Access 2000 格式

在 Access 2003 中，除了可以打开和转换旧版本的数据库外，还可以将 Access 2003 数据库转换为 Access 97 或 Access 2000 格式的数据库，具体操作步骤如下：

- (1) 打开需要转换的 Access 2003 数据库。
- (2) 选择“工具”菜单中的“数据库实用工具”命令，从“数据库实用工具”级联菜单中选择“转换数据库”命令，出现转换格式选择级联菜单，然后选择转换格式。
- (3) 在“将数据库转换为”对话框中确定转换后数据库的存放位置、类型以及名称。
- (4) 单击“保存”按钮，即可将当前数据库转换为 Access 97 或 Access 2000 数据库。

 注意：在转换一个数据库之前，应该先对原数据库进行备份。

4.5.3 将旧版本的Access数据库转换为Access 2003 数据库

将旧版本数据库转换为 Access 2003 数据库的主要操作步骤如下：

- (1) 关闭已经打开的数据库。
- (2) 选择“工具”菜单中的“数据库实用工具”命令，从“数据库实用工具”级联菜单中选择“转换数据库”命令，再从“转换数据库”级联菜单中选择“转为 Access 2002—2003 文件格式数据库版本”命令，系统将打开“数据库转换来源”对话框。
- (3) 在对话框中指定需要转换的数据库路径、文件类型以及文件名。
- (4) 单击“转换”按钮，打开“将数据库转换为”对话框。
- (5) 在“将数据库转换为”对话框中指定转换后数据库存放的位置、类型以及文件名。
- (6) 单击“保存”按钮，Access 即可对选定的数据库进行转换。

 注意：在转换带有链接表的旧版本 Access 数据库时，用户应该确保链接表位于原始文件夹中，如果 Access 找不到链接的表，转换操作将无法进行。

习 题

1. 新建数据表的方法有几种？
2. 在 Access 2003 中，数据字段的类型分为哪几种？
3. 什么是主码？在表中使用主码有哪些好处？
4. 使用“设计视图”创建一个表，并为表指定一个主码。
5. 设置表中字段的“常规”和“查阅”属性，并在属性中练习使用组合框。
6. 参照完整性规则是什么？在什么情况下使用？
7. 索引的作用是什么？如何建立索引？
8. 什么是筛选？筛选包含哪些内容？各有何不同作用？
9. 什么是数据的导入与导出？主要操作步骤有哪些？
10. 通过不同的方式创建学生表、课程表和成绩表并建立三个表之间的关系。

第 5 章 查询操作与高级应用

本章学习目标

可利用查询可以选择一组满足指定条件的记录，还可以将不同表中的信息组合起来，提供一个相关数据项的统一视图。本章从查询的种类与应用着手，对查询的建立方法、查询条件、查询设计、SQL 查询以及查询操作进行专门的介绍。通过本章的学习，读者应该掌握以下内容：

- 认识 Access 2003 中的查询种类
- 掌握不同查询的建立方法
- 查询准则的设计
- SQL 查询语句的使用
- 使用查询操作表或表中数据

5.1 查询的种类与应用

查询是 Access 处理和分析数据的工具，它能够把多个表中的数据抽取出来，供用户查看、更改和分析使用。

5.1.1 查询的种类

Access 2003 支持 5 种查询方式：选择查询、操作查询、SQL 查询、交叉表查询和参数查询。下面简单介绍各种查询的特点。

1. 选择查询

选择查询是最常见的查询类型。选择查询可以从一个或多个表或者其他的查询中获取数据，并按照所需要的排列次序显示，利用选择查询可以方便地查看一个或多个表中的部分数据。查询的结果是一个数据记录的动态集，用户可以对动态集中的数据记录进行修改、删除，也可以增加新的记录，对动态集所作的修改会自动写入相关联的表中。

2. 操作查询

操作查询就是在一个操作中对查询中所生成的动态集进行更改的查询。操作查询可分为：生成表查询、追加查询、更新查询和删除查询，生成表查询和追加查询可以复制原有的数据，更新查询和删除查询可以更改现存的数据。操作查询只能更改和复制用户的数据，而不能返回数据记录。

- 生成表查询：可以利用从一个或多个表及查询中的查询结果创建一个新表。
- 追加查询：将查询结果添加到现存的一个或多个表或者查询的末尾。
- 更新查询：根据查询中指定的条件，更改一个或多个表中的记录。
- 删除查询，根据查询中指定的条件，从一个或多个表中删除相关记录。

3. SQL查询

SQL 是一种结构化查询语言，是数据库操作的工业化标准语言，使用 SQL 语言可以对任何数据库管理系统进行操作。SQL 查询就是使用 SQL 语言创建的查询，它又可以分为联合查询、传递查询和数据定义查询等。其中联合查询是 SQL 语言所特有的，可以将来自一个或多个表或者其他查询中的字段组合起来作为查询结果中的一个字段，可以创建快照类型的记录集合对象；传递查询就是直接将命令发送到 ODBC 数据源的查询，在服务器上进行查询操作；数据定义查询也是 SQL 查询所专有的，用户可以通过 SQL 语句来创建、修改或删除表对象，并且可以动态地对表的结构进行修改。

4. 交叉表查询

交叉表查询可以汇总数据字段的内容，例如日期或数字字段。在这种查询中，汇总计算的结果显示在行与列交叉的单元格中。交叉表查询还可以计算平均值、总计、最大值或最小值等。

5. 参数查询

参数查询可以在运行查询的过程中输入参数值来设定查询准则，而不必重新创建一个新查询。参数查询不是一种独立的查询，它扩大了其他查询的灵活性。执行参数查询时，系统会显示一个对话框提示用户输入参数的值。

5.1.2 查询的应用

查询的应用主要有如下几个方面。

(1) 选择字段

在查询中，可以只选择表中的部分字段。例如，对于学生表，只选择学号、姓名、性别、系别和专业建立一个查询。利用这一查询功能，可以通过选择一个表中的不同字段生成所需的多个表。

(2) 选择记录

根据指定的条件查询表中的记录。例如，在学生表中，可以创建一个只显示专业为“计算机科学与技术”的学生信息。

(3) 编辑记录

编辑记录主要是添加记录、修改记录和删除记录等。可以利用查询添加、修改和删除表中的记录。例如，根据入学日期，将已毕业的学生从学生表中删除。

(4) 实现计算

可以在查询中进行各种统计计算，如计算某门课的平均成绩。还可以建立一个计算字

段来保存计算结果。

(5) 利用查询的结果生成窗体或报表

要从一个或多个表中选择合适的数据显示在窗体和报表中，可以先基于一个或多个表创建查询，然后，以查询的结果作为窗体或报表的数据源。

(6) 利用查询的结果创建表

利用查询的结果可以建立一个新表。例如，可以利用查询将学生表中的“系别”为“计算机系”的学生记录存放到一个新表中。

总之，用户可以利用查询查看指定的字段，显示特定条件的记录。如果需要，可以将查询的结果保存起来，也可以将其作为窗体或报表的数据来源。

5.2 查询的建立方法

创建查询的方法主要有使用查询向导和查询设计视图两种。使用查询向导是利用提示指导用户完成创建查询的工作，而使用查询设计视图，不仅可以完成查询设计，也可以修改已有的查询，而且设计视图的功能更丰富和更强大。

5.2.1 使用查询向导

简单查询是应用最广泛的一种查询，它可以从一个或多个表、查询中查找相关记录。使用“简单查询向导”创建的查询具有以下特点：

- 不能添加选择准则或者指定查询的排序次序。
- 不能改变查询中字段的次序，字段将一直以向第一个向导对话框中添加它们时的顺序出现。
- 如果所选的字段中有一个或者多个数字字段，该向导允许放置一个汇总查询，显示数字字段的总计值、平均值、最小值或者最大值。在查询结果集中还可以包含一个记录数量的计数。
- 如果所选的一个或者多个字段为“日期/时间”数据类型，则可以指定按日期范围分组，汇总查询一天、一月、一季或者一年。

1. 简单查询向导

【例 5.1】 以学生、课程和成绩表为数据源，利用向导创建学生成绩查询。

主要操作步骤如下：

(1) 在数据库窗口中，选择“查询”对象，双击“使用向导创建查询”，系统打开“简单查询向导”的第一个对话框，如图 5.1 所示。

(2) 在“表/查询”下拉列表中选择“表：学生”，学生表的所有字段都将出现在“可用字段”列表中。

(3) 在“可用字段”列表中选择查询字段，例如，选择学号、姓名字段并单击  按钮将其添加到“选定的字段”列表中，也可以通过双击字段将其添加到“选定的字段”列表。

重复步骤(2)和(3)，完成“课程”表的“课程名称”和“成绩”表的“成绩”字段的添加。

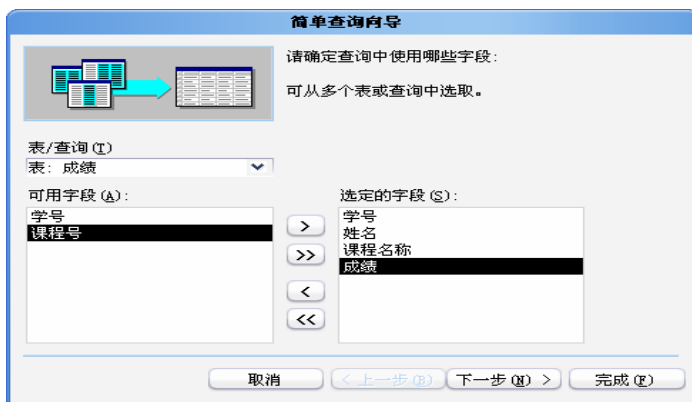


图 5.1 “简单查询向导”对话框之一

(4) 单击“下一步”，打开第二个向导对话框，选择是使用明细查询还是使用汇总查询，明细查询可以显示每个记录的每个字段，汇总查询可以计算字段的总值、平均值、最小值、最大值和记录数等。接受默认的“明细”选项，如图 5.2 所示。

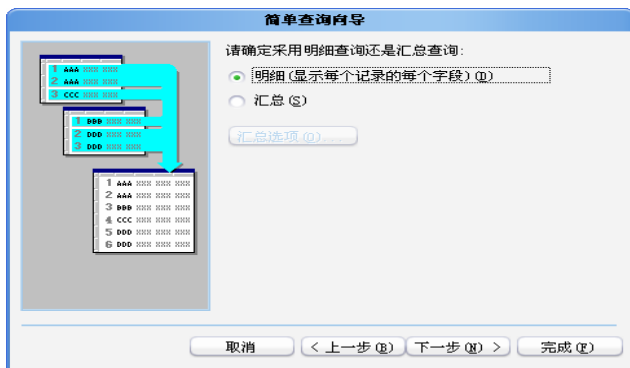


图 5.2 “简单查询向导”窗口之二

(7) 单击“下一步”按钮，打开最后一个对话框，如图 5.3 所示。

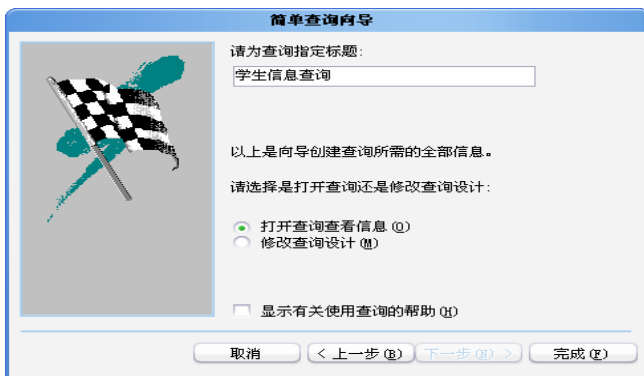


图 5.3 “简单查询向导”窗口之三

在“请为查询指定标题”文本框中,输入查询名字“学生成绩查询”,单击“完成”,在“数据表”视图中显示查询结果,如图 5.4 所示。

学号	姓名	课程名称	成绩
30203301	杜少华	微机原理	81
30203301	杜少华	计算机多媒体技术	93
30203301	杜少华	数据结构	90
30203301	杜少华	Access 2003数据库	78
30203301	杜少华	大学英语	76
30203302	张大伟	大学英语	47
30203302	张大伟	微机原理	69
30203302	张大伟	计算机多媒体技术	74
30203302	张大伟	数据结构	77
30203302	张大伟	Access 2003数据库	64
30203303	王朋立	大学英语	75
30203303	王朋立	Access 2003数据库	61
30203303	王朋立	数据结构	97
30203303	王朋立	微机原理	92
30203303	王朋立	计算机多媒体技术	65
30203305	陈小东	大学英语	81

图 5.4 查询结果

2. 向导的汇总查询

【例 5.2】 查询每位同学的选课门数、总成绩、平均成绩、最低成绩和最高成绩。
主要操作步骤如下:

(1) 在“表/查询”下拉列表框中选择“表: 学生”,添加学生表的“姓名”字段到“选定的字段”列表中;在“表|查询”下拉列表框选择“表: 成绩”,添加成绩表的“成绩”字段到“选定的字段”列表中,如图 5.5 所示。

图 5.5 选择查询字段

注意: 汇总查询必须包含用于分组数据的字段(姓名)和汇总的数字值。如果添加了其他字段,例如“课程名称”,则所有记录都将出现在汇总查询中,将无法得到所期望的汇总。

(2) 单击“下一步”按钮,在打开的对话框中,选择“汇总”选项,单击“汇总选项”按钮,打开“汇总选项”对话框,如图 5.6 所示。选中“汇总”、“平均”、“最小”和“最大”复选框,分别计算学生的总成绩、平均成绩、最低成绩和最高成绩。选中“统计成绩中的记录数”复选框,为分组查询添加一列,提供记录计数。

(3) 单击“确定”按钮，返回第二个向导对话框，然后单击“下一步”按钮，转向第三个向导对话框，为查询命名。

(4) 单击“完成”按钮，执行该汇总查询，查询结果如图 5.7 所示。

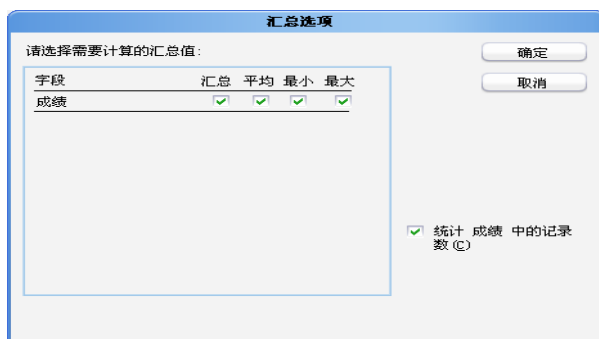
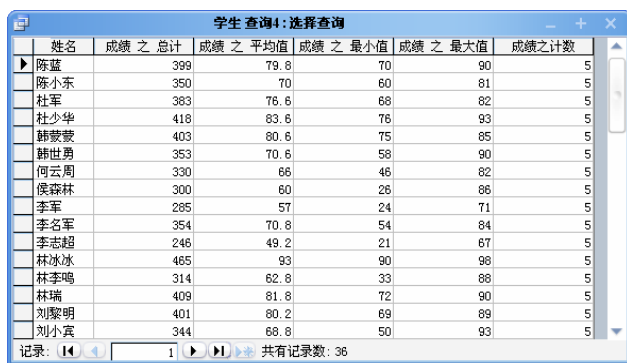


图 5.6 “汇总选项”对话框



姓名	成绩之 总计	成绩之 平均值	成绩之 最小值	成绩之 最大值	成绩之计数
陈蓝	399	79.8	70	90	5
陈小东	350	70	60	81	5
杜军	383	76.6	68	82	5
杜少华	418	83.6	76	93	5
韩蒙蒙	403	80.6	75	85	5
韩世勇	353	70.6	58	90	5
何云周	330	66	46	82	5
侯森林	300	60	26	86	5
李军	285	57	24	71	5
李名军	354	70.8	54	84	5
李志超	246	49.2	21	67	5
林冰冰	465	93	90	98	5
林李鸣	314	62.8	33	88	5
林瑞	409	81.8	72	90	5
刘黎明	401	80.2	69	89	5
刘小宾	344	68.8	50	93	5

图 5.7 汇总查询结果

5.2.2 使用查询设计器

使用“简单查询向导”有很大的局限性，最好的查询方法是在 Access 2003 的图形化“查询设计”窗口中设计查询。查询“设计视图”是 Access 最为强大的功能特征之一。

在 Access 2003 中，查询有 3 种视图：设计视图、数据表视图和 SQL 视图。使用设计视图，不仅可以创建各种类型的查询，而且可以对已有的查询进行修改。例如，要打开利用向导创建的“学生成绩查询”，可在数据库窗口中选中该查询，然后单击工具栏上的“设计”按钮，结果如图 5.8 所示。

1. 查询设计视图

设计视图的窗口分两部分，上半部分显示查询所使用的表对象，下半部分定义查询设计的表格。

- 字段：选择查询中要包含的表字段。
- 表：选择字段的来源表。
- 排序：定义字段的排序方式。

- 显示：设置是否在数据表视图中显示所选字段。
- 条件：设置字段的查询条件。
- 或：用于设置多条件之间的或条件。

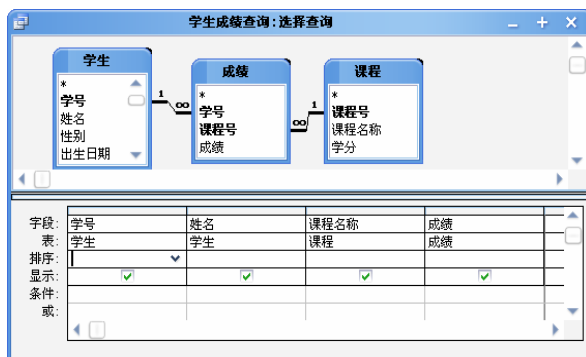


图 5.8 查询设计视图

2. 查询设计视图的工具栏

查询设计视图的工具栏如图 5.9 所示。

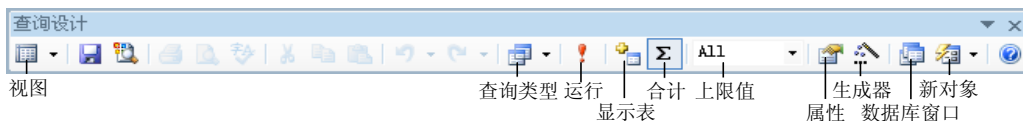


图 5.9 设计视图工具栏

- 视图：可以在查询的 3 个视图之间切换。
- 查询类型：可以在选择查询、交叉表查询、生成表查询、更新查询、追加查询和删除查询之间切换。
- 运行：执行查询，以数据表视图的形式显示结果。
- 显示表：打开显示表对话框，列出当前数据库中所有的表和查询，用户可以选择查询要使用的对象。
- 合计：可以对在查询设计中增加“总计”行，用于进行各种统计计算。
- 上限值：可以查询结果的显示进行约定，并在文本框中指定所要显示的范围。
- 属性：显示光标指向的对象属性，在这里对字段属性进行修改，不会反映到数据表中。
- 生成器：用于使用表达式生成器对话框，生成查询条件表达式，该按钮当光标处于“条件”栏中时才有效。
- 数据库窗口：返回数据库窗口。
- 新对象：打开新建表、新建查询和新建窗体等对话框，以创建相应的对象。

3. 创建查询

在设计视图中创建查询，首先应单击“显示表”按钮，在打开的“显示表”对话框中选择查询所依据的表、查询，并将其添加到设计视图的窗口中。如果选择多个表，多个表

之间应先建立关联。

【例 5.3】 利用学生、课程和成绩表，创建一个不及格学生的成绩查询。

主要操作步骤如下：

- (1) 在“显示表”对话框中，把学生、课程和成绩表添加到设计视图的窗口中。
- (2) 依次拖入或在“字段”行中选择“姓名”、“课程名称”和“成绩”字段。
- (3) 在成绩列中的“条件”行中输入条件“<60”，设计结果如图 5.10 所示。

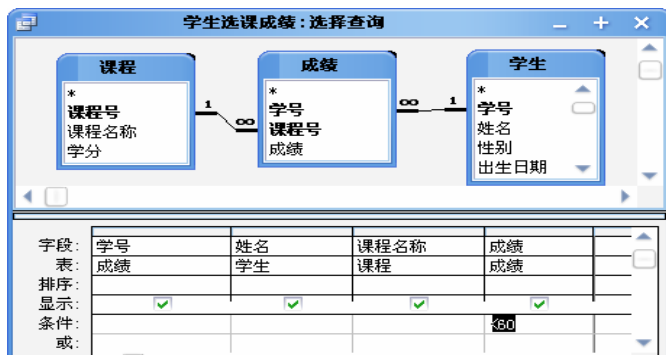


图 5.10 学生选课成绩查询设计视图

- (4) 单击工具栏上的“运行”按钮，查询结果如图 5.11 所示。

学号	姓名	课程名称	成绩
30203302	张大伟	大学英语	47
30203306	王云	微机原理	59
30203306	王云	计算机多媒体技术	58
30203308	郑玉栋	数据结构	53
30203308	郑玉栋	大学英语	39
30203310	朱云波	微机原理	27
30203310	朱云波	数据结构	21
30203311	宋名	大学英语	52
30203311	宋名	微机原理	57
30203311	宋名	数据结构	56
30203312	王开开	微机原理	45

图 5.11 学生选课成绩查询运行结果

如果生成的查询不满足要求，可以单击视图按钮，返回设计视图进行修改。关闭设计视图窗口时，系统会要求输入查询名称，可以选择一个符合查询特征的名字，以保存查询对象，便于以后的使用。

5.3 查询条件

在实际应用中，用户的查询并非只是简单的查询，往往需要指定一定条件，使查询结果中仅包含满足查询条件的记录。例如，查找计算机系的张姓学生，这种带条件的查询需要通过设置查询条件来实现。查询条件是运算符、常量、字段值、函数以及字段名和属性等的任意组合，能够计算出一个结果。了解查询条件的组成及书写方法，对建立条件查询

非常重要。

前面介绍的查询中，查询方式是根据查询准则检查表中的字段值是否相匹配，但在实际查询中往往需要对原有的数据进行适当的加工，以便显示出实际需要的结果。例如在“学生期末考试成绩表”中，要查询每名学生的平均成绩、总评成绩、单科的班级平均成绩等，就需要使用表达式与函数进行计算，除此之外，数据库的其他对象也经常要用到表达式。

5.3.1 表达式

要想使用 Access 2003 数据库中的表达式，首先要清楚什么是常数、常量、变量和如何使用 Access 2003 的表达式。

1. 常数

通常可以在表达式中使用常数，也可以在用来编写 Access 模块的 VBA 语句中将它赋给常量和变量。在具体应用中，日期型的常数需要用界限符(#)括起来，例如，#2003-10-10#；字符型常数要用单引号(')或双引号(")括起来(单引号和双引号必须是半角的符号)；数值型常数可以直接引用值，例如，123.56、123 等。

2. 常量

常量代表固定不变的数值或字符串值。例如，True、False 和 Null 都是常量。常量可以代表单个字符串、数值、任何包含数值或逻辑运算符的表达式，但不能包含 IS 语句和乘幂运算符。在表、查询、窗体、报表及宏的表达式中均可以使用常量，在模块中需要使用 Const 语句来声明常量。常量的命名规则与变量的命名规则相同。

Access 系统支持 3 类常量：

- 符号常量：用 Const 语句说明，可以在模块中引用。
- 固有常量：是 Access 系统自动定义的作为操作参数和函数值的常量，并提供了对 VB、VBA、DAO 库常量的访问。
- 系统定义常量：True、False 和 Null。

3. 变量

变量是指命名的存储空间，用于存储在程序执行过程中可以改变的数据。有关变量的详细内容将在第 10 章介绍。

4. 表达式

用运算符将常数、常量、变量、函数以及字段名、控件和属性等连接起来的式子称为表达式，表达式将计算出一个单个值。可以将表达式作为许多属性和操作参数的设置值；可以利用表达式在查询中设置条件(搜索条件)或定义计算字段；还可以利用表达式在窗体、报表和数据访问页中定义计算控件，以及在宏中设置条件。在 Access 2003 中，可以使用表达式来定义检查约束。

Access 2003 系统提供了算术运算、关系(比较)运算、连接运算和逻辑运算 4 种基本运

算表达式。

(1) 算术表达式

算术运算符, 包括+、-、*、/, 也就是常用的四则运算符。算术运算符仅用于数值运算并且必须有两个数值运算数。

在查询中使用的计算字段就需要用到算术运算表达式。

【例 5.4】 利用“学生期末考试成绩表”查询每位同学的总成绩和平均成绩。

主要设计步骤如下:

- ① 打开“学生”数据库, 进入查询的“设计视图”。
- ② 在“显示表”对话框中的“表”选项卡中选择“学生期末考试成绩表”, 单击“添加”按钮后, 再单击“关闭”按钮。
- ③ 在表设计视图中, 把表中的学号和姓名字段直接拖到字段行中。
- ④ 还要在字段行中加入两列, 一列计算平均成绩, 一列计算总成绩, 由于总成绩和平均成绩并不是表中字段, 在此要使用如下格式定义:

总成绩: (高等数学+大学英语+普通物理+高级语言程序设计+微机原理)

平均成绩: (高等数学+大学英语+普通物理+高级语言程序设计+微机原理)/5

查询的设计视图如图 5.12 所示。



图 5.12 设计视图

- ⑤ 选择“查询”菜单中的“运行”命令, 即显示查询结果, 如图 5.13 所示。

学号	姓名	总成绩	平均成绩
30203304	张源源	422	84.4
30203334	张照正	418	83.6
30203313	何云周	418	83.6
30203326	杜军	418	83.6
30203305	陈小东	415	83
30203331	朱谦	408	81.6
30203335	林瑞	398	79.6
30203308	郑玉栋	397	79.4
30203338	韩世勇	393	78.6
30203325	陈蓝	392	78.4
30203329	周玲	390	78

图 5.13 查询结果

(2) 关系表达式

关系运算符包括: >、<、>=、<=、<>。用关系运算符连接的两个表达式构成关系表达

式, 结果为一个逻辑值 True 或者 False。

(3) 连接运算表达式

连接运算符包括: &和+。连接运算符具有连接字符串的功能, 详细介绍见第 10 章。

(4) 逻辑运算表达式

逻辑运算符包括: And、Or、Not。逻辑运算主要用于对真、假判断。And 表示两个操作数都为 True 时表达式的值才为 True; Or 表示两个操作数只要有一个为 True, 表达式的值就为 True; Not 表示取操作数的相反值。

例如, 查询计算机系的学生党员与查询计算机系学生及全院学生党员, 可分别设定条件为:

政治面貌="党员" And 系别="计算机系"

政治面貌="党员" Or 系别="计算机系"

【例 5.5】 在学生表中查询不是男生的记录。

可以在“性别”字段的条件行中输入:

Not -1(等价于<>-1)

因为性别字段的类型为“是/否”型, -1 表示“是”, 我们约定为男; 0 表示“否”, 我们约定为女。

(5) 其他运算表达式

其他的 Access 2003 运算符与比较运算有关, 这些运算符根据字段中的值是否符合这个运算符的限定条件返回 True 或 False。

- Between...And...

用于指定一个字段值的取值范围, 指定的范围之间用 And 连接。

【例 5.6】 在学生期末考试成绩表中, 查找所有“大学英语”成绩在 70~90 之间的学生。

可以在“大学英语”字段条件行中输入:

Between 70 And 90(等价于>=70 And <=90=

【例 5.7】 查询 1981 年出生的学生信息。

可以在出生日期字段的条件行中输入:

Between #1981-01-01# and #1981-12-31#

或

>= #1981-01-01# And <=# 1981-12-31#



注意: #号是日期型数据的定界符, 这样系统就不会将该数据认为是文本或其他数据类型。

- In

用于指定一个字段值的列表, 列表中的任何一个值都可与查询的字段相匹配。

【例 5.8】 查询所有政治面貌为党员或团员的学生。

可以在政治面貌字段的条件行中输入:

In("党员","团员")

或

"党员" Or "团员"

上面两种用法是等价的,当表达式中包含的值较多时,使用 In 运算符会使表达式更简洁,意义也更明确。

- Is

指定所在字段中是否包含数据,Is Null 表示查找该字段没有数据的记录,Is Not Null 表示查找该字段有数据的记录。

【例 5.9】 查询所有大学英语无成绩的学生。

可以在大学英语字段的条件行中输入:

Is Null

- Like

查找相匹配的文字,用通配符来设定文字的匹配条件。

Access 2003 提供的通配符有:

- ◆ ? 代表任意一个字符。
- ◆ * 代表任意多个字符。
- ◆ # 代表任意一个数字位(0~9)。
- ◆ [字符表] 代表在字符表中的单一字符。
- ◆ [!字符表] 代表不在字符表中的单一字符。

可以使用一对方括号为字符串中该位置的字符设置一个范围,如[0-9]、[a-z]、[!a-z]等,连接符“-”用于分隔范围的上下界。

例如:

Like "AV?? " 以 AV 开始的连续 4 个字符,后 2 位为任意字符

Like "李*" 表示以李开头的所有字符串

Like "A[b-f]V### " 以 A 开头,后跟 b~f 之间的一个字母和 3 个数字

Like "?[!3-8][A-P]*" 第一个字符为任意字符,第二个为非 3~8 的任意字符,第三个为 A~P 之间的一个字母,其后为任意字符串

5.3.2 标准函数

表达式不但可以使用数学运算符,还可以使用 Access 2003 内置函数。系统提供了大量的标准函数,如数值函数、字符函数、日期时间函数和统计函数等。这些函数为用户设计查询条件提供了极大的便利,也为进行统计计算、实现数据处理提供了有效的方法。

关于函数的详细内容将在第 10 章中介绍,在此仅简单介绍常用的统计函数。

1. 求和函数

格式: SUM(<字符串表达式>)

功能: 返回字段中值的总和。

说明：“字符串表达式”可以是一个字段名(数值类型)，或者是含有数值类型字段的表达式。

例如，有一个“单价”字段和一个“数量”字段，求总价。可以用 SUM 函数计算其总价：SUM(单价*数量)

2. 求平均函数

格式：AVG(<字符串表达式>)

功能：求数值类型字段的平均值。

说明：

① “字符串表达式”可以是一个字段名(数值类型)，或者是含有数值型字段的表达式。

② AVG 不计算任何 Null 值字段。

例如，设有一个“成绩”字段，可用 AVG(成绩)求平均分。

3. 统计记录个数函数

格式：COUNT(<字符串表达式>)

功能：统计记录个数。

说明：

① “字符串表达式”可以是一个字段名(数值类型)，或是含有数值型字段的表达式。

② 当用格式 COUNT(*)时，将统计所有记录的个数，包括有 Null 值字段的记录。

例如，COUNT(姓名)(其中“姓名”为字段名)。统计姓名字段不是 Null 值的所有记录的个数。

4. 最大、最小值函数

格式：MAX(<字符串表达式>)

MIN(<字符串表达式>)

功能：返回一组指定字段中的最大、最小值。

说明：“字符串表达式”可以是一个字段名(数字类型)，或者是含有数字型字段的表达式。

例如，有一个“成绩”字段，可以用 MAX(成绩)求该字段中的最大值，用 MIN(成绩)求该字段中的最小值。

在查询条件设计过程中，灵活运用上面介绍的函数与表达式，对于增强查询的功能、丰富查询的应用非常重要。

5.4 查询设计

在实际应用中，需要创建的查询多种多样，本节将重点介绍条件选择查询、交叉表查询、参数查询及 4 种操作查询的设计。

5.4.1 条件选择查询

使用 Access 2003 提供的向导可以方便地创建基于一个或多个表的不带条件的查询。但在日常应用中,需要建立的查询往往带有一定的条件,例如,查找计算机系 2004 年入学的计算机科学与技术专业的学生。这种查询需要通过“设计”视图来创建,在“设计”视图的“条件”行输入查询条件。

【例 5.10】 查询计算机系 2004 年入学的计算机科学与技术专业的学生,显示学号、姓名、性别、入学日期和系别。

主要操作步骤如下:

(1) 在“学生管理”数据库窗口中,单击“查询”对象,然后双击“在设计视图中创建查询”项,系统将打开如图 5.14 所示的查询“设计”视图,同时在此视图上面还出现一个“显示表”对话框。

(2) 在“显示表”对话框中,把学生表添加到设计视图上半部分的窗口中。

(3) 分别双击“学号”、“姓名”、“性别”、“入学日期”和“系别”等字段,这时 5 个字段将依次显示在“字段”行上的第 1 列到第 5 列中,同时,“表”行将显示出这些字段所在表的名称,“显示”行复选框均选中,代表查询结果中显示这些字段。但由于查询要求显示的字段并没有“专业”字段,该字段仅作为查询的一个条件,所以应取消“专业”字段“显示”行上的复选框的选中状态,结果如图 5.15 所示。

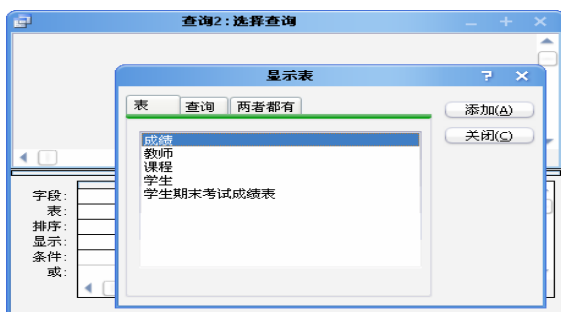


图 5.14 查询“设计”视图



图 5.15 设置查询字段

(4) 在“入学日期”字段列的“条件”行中可以输入以下等价条件:

Year([入学日期]) = 2004

或

Between #2004-1-1# And #2004-12-31#

或

>= #2004-1-1# And <= #2004-12-31#

在“系别”和“专业”字段列的“条件”行中分别输入条件:“计算机系”和“计算机科学与技术”,设计结果如图 5.16 所示。

 **注意:** 在查询表达式中使用字段名,要用方括号括起来,运算符与引号等要用西文字符。

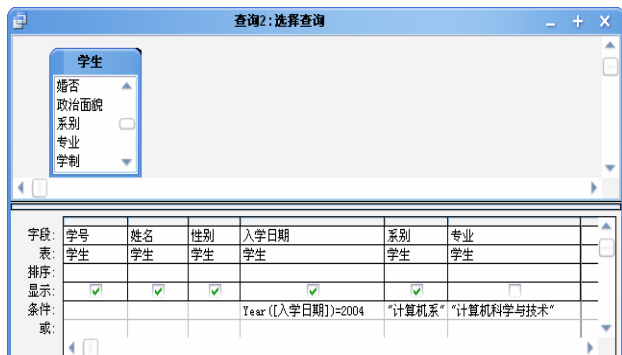


图 5.16 学生信息查询设计视图

(5) 单击工具栏上的“保存”按钮,在弹出的“另存为”对话框中,命名查询为“计算机系计算机科学与技术专业 2004 年入学的学生”,然后单击“确定”按钮,保存创建的查询。可以用多种方式运行该查询,查看查询结果。

设计查询,要正确理解查询条件。在上面建立的查询中,涉及 3 个条件,查询结果要求同时满足 3 个条件,即是与(and)条件,所以条件均写在“条件”行上。若希望查询计算机科学与技术专业或者 2004 年入学的学生,就要使用“或”行。

例如,查询计算机科学与技术专业或者 2004 年入学的学生,查询设计视图结果如图 5.17 所示。

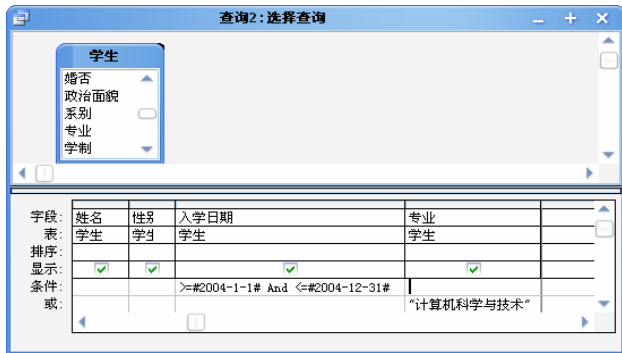


图 5.17 使用“或”条件

如果生成的查询不满足要求，可以单击视图按钮，返回设计视图进行修改。

5.4.2 交叉表查询

交叉表查询以行和列的字段作为标题和条件选取数据，并在行与列的交叉处对数据进行汇总、统计计算等。例如，我们曾经依据学生、课程和成绩表建立了一个“选课成绩查询”，在这个查询中给出了每名学生选课的成绩，但由于在“课程名称”列中出现了重复的课程名称，使查询的结果不够清晰和合理，使用交叉表查询可以很好地解决这一问题。

设计交叉表查询，用户需要指定三种字段：一是放在数据表最左端的行标题，它把某一字段或相关的数据放入指定的一行中；二是放在数据表最上边的列标题，它对每一列指定的字段或表进行统计，并将统计结果放在该列中；三是放在数据表行与列交叉位置上的字段，用户需要为该字段指定一个总计项，如 Sum、Avg、Count 等。对于交叉表查询，用户只能指定一个总计类型的字段。

1. 使用向导创建交叉表查询

【例 5.11】 建立学生选课成绩交叉表查询。

主要操作步骤如下：

(1) 在“数据库”窗口中单击“查询”对象，再单击“新建”按钮，在“新建”查询对话框中选择“交叉表查询向导”，打开如图 5.18 所示交叉表查询向导对话框之一。在“视图”选项组中选择查询，在上方的列表框中选择“学生选课成绩”查询。

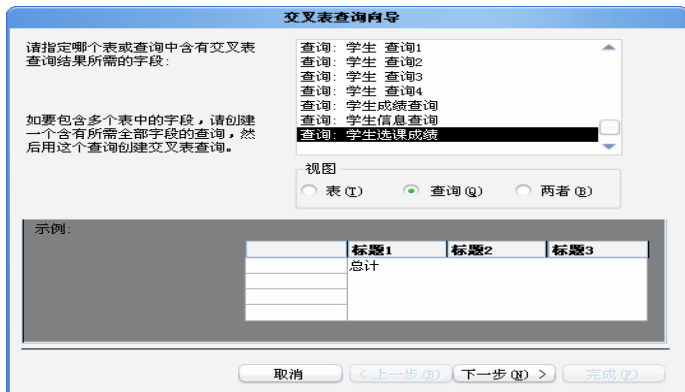


图 5.18 交叉表查询向导对话框之一

(2) 单击“下一步”按钮，在向导对话框之二中(如图 5.19 所示)，为交叉表选择一个行标题，如“姓名”；单击“下一步”按钮，在向导对话框之三(如图 5.20 所示)，为交叉表选择一个列标题，如“课程名称”；单击“下一步”按钮，打开向导对话框之四，如图 5.21 所示。

(3) 为行和列的交叉点指定一个值。因为要显示学生选课的成绩，在“字段”列表框中选中“成绩”；在函数列表框中选择一个总计函数，由于所建交叉表只需显示成绩，不需作其他计算，所以选择“第一项”，并取消选择“是，包含各行小计”复选框。



图 5.19 交叉表查询向导对话框之二

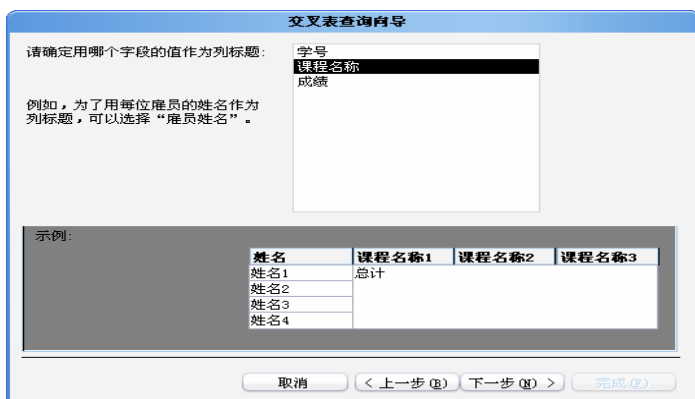


图 5.20 交叉表查询向导对话框之三

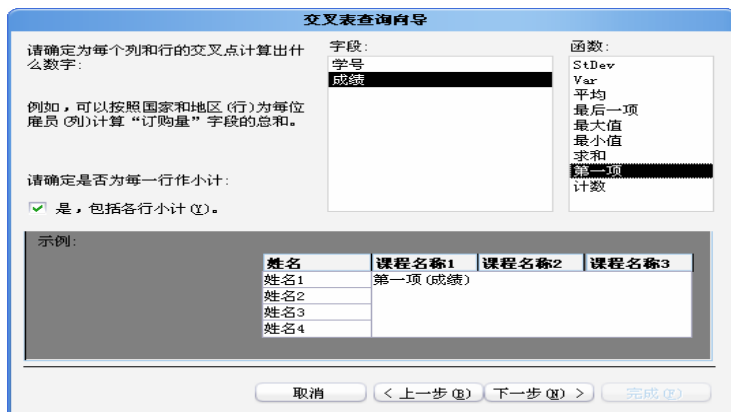


图 5.21 交叉表查询向导对话框之四

(4) 为所创建的交叉表查询指定一个名字,如“交叉表成绩查询”,单击“完成”按钮,创建交叉表查询的运行结果如图 5.22 所示。

这里应注意:创建交叉表的数据源必须来自一个表或查询。如果数据源来自多个表,可以先建立一个查询,然后再用此查询作为数据源。

姓名	Access 2003数据库	大学英语	计算机多媒体技术	数据结构	微机原理
陈蓝	70	83	90	80	76
陈小东	61	81	70	78	60
杜军	76	81	76	82	68
杜少华	78	76	93	90	81
韩蒙蒙	81	79	85	83	75
韩世勇	87	79	90	58	59
何云周	76	82	76	46	50
侯森林	56	68	86	26	64
李军	61	63	71	24	66
李名军	66	82	84	54	68
李志超	50	64	67	21	44
林冰冰	90	92	90	98	95
林李鸣	65	67	88	33	61
林瑞	80	90	80	87	72
刘敬田	78	81	84	89	89

记录: 12 共有记录数: 36

图 5.22 交叉表查询运行结果

2. 在“设计”视图下创建交叉表查询

使用设计视图，可以基于多个表创建常见交叉表查询。

【例 5.12】 在“学生管理”数据库中，使用“学生”、“课程”和“成绩”表创建一个交叉表查询，使其显示每位学生各门课的成绩。

主要操作步骤如下：

(1) 在“学生管理”数据库窗口中，单击“查询”对象，然后双击“在设计视图中创建查询”项，系统将打开查询设计视图，同时显示“显示表”对话框。

(2) 在“显示表”对话框中，把“学生”、“课程”和“成绩”表添加到设计视图上半部分的窗口中，然后关闭“显示表”对话框。

(3) 分别双击学生表中的“姓名”字段、课程表中的“课程名称”字段和成绩表中的“成绩”字段，依次添加到“字段”行的第 1 列到第 3 列。

(4) 单击工具栏上的“查询类型”按钮，然后从下拉列表中选择“交叉表查询”命令。

(5) 单击“姓名”列的“交叉表”行单元格，选择其下拉列表中的“行标题”选项，使“姓名”放在每行的左边；单击“课程名称”列的“交叉表”行单元格，选择其下拉列表中的“列标题”选项，使“课程名称”放在第一行上；单击“成绩”列的“交叉表”行单元格，选择其下拉列表中的“值”选项，使在行列的交叉处显示成绩数值；单击“成绩”列的“总计”行单元格，选择其下拉列表中的“第一条记录”函数，设计视图结果如图 5.23 所示。

字段:	姓名	课程名称	成绩
表:	学生	课程	成绩
总计:	分组	分组	第一条记录
交叉表:	行标题	列标题	值
排序:			
条件:			

图 5.23 交叉表设计视图

(6) 保存与运行创建的交叉表查询，结果与图 5.22 相同。

5.4.3 参数查询

参数查询可以在运行查询的过程中根据参数输入值自动设定查询的规则，用户在执行参数查询时会显示一个输入对话框以提示用户输入信息，这种查询叫做参数查询。当需要对某个字段进行参数查询时，首先切换到这个查询的设计视图，然后在作为参数使用的字段下的“条件”单元格中的方括号内输入相应的提示文本，此查询运行时，Access 2003 将显示该提示。

参数查询是建立在选择查询或交叉查询的基础之上的，如果用户希望根据某个或某些字段的值来查找记录，就可以使用参数查询。在参数查询中，可以建立单参数查询，也可以建立多参数查询。

【例 5.13】 以学生、课程和成绩表为数据源，查询某班级、某门课和某分数之上的学生选课情况。假定学号的第 4、5 位为班级号码。

主要操作步骤如下：

(1) 在“学生管理”数据库窗口中，单击“查询”对象，然后双击“在设计视图中创建查询”，打开查询设计视图，同时显示“显示表”对话框。

(2) 在“显示表”对话框中，把“学生”、“课程”和“成绩”表添加到设计视图上半部分的窗口中，然后关闭“显示表”对话框。

(3) 分别双击学生表中的“学号”和“姓名”字段、课程表中的“课程名称”字段和成绩表中的“成绩”字段，依次添加到“字段”行的第 1 列到第 4 列。

(4) 在“字段”行的第 1 列单元格中，把“学号”字段名改为：班级: Mid([学生].[学号],4,2)，在“条件”行单元格输入：[输入班级:]。

其含义是：在查询结果中，显示别名“班级”，取每个记录“学号”字段值的第 4、5 位与“输入班级”参数值进行比较，确定是否是要查询的班级。

(5) 在“课程名称”列的“条件”单元格输入：[输入课程名称:]。

(6) 在“成绩”列的“条件”单元格输入：>[输入成绩(>=):]。

设计视图结果如图 5.24 所示。

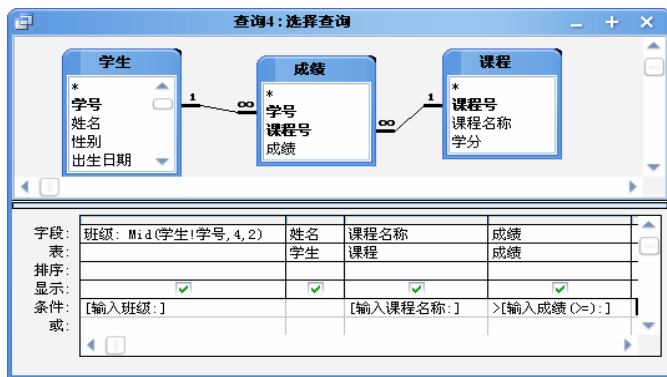


图 5.24 参数查询设计视图

(7) 单击工具栏上的“视图”按钮，或单击工具栏上的“运行”按钮，系统将依次显示输入班级、输入课程名称和输入成绩 3 个“输入参数”对话框，用户可以根据需要输入

参数值, 查询需要的学生所选课的成绩信息。

 **注意:** 在本例中, 学号的第 4, 5 位代表班级号, 为取得班级号, 用到了字符串截取函数 Mid, 可参见第 10 章介绍。

对于参数查询, 关键是查询条件含义的理解, 较之选择查询, 参数查询能满足更多的应用要求。

5.4.4 操作查询

操作查询用于创建新表或者修改现有表中的数据。Access 2003 提供的操作查询有以下几种类型:

- 生成表查询: 用查询结果集中的数据创建新表。从表中查询数据要比从查询中访问数据快, 如果经常需要从多个表中提取数据, 最好的办法是使用生成表查询, 即从多个表中提取数据组合起来生成一个新表永久保存。
- 删除查询: 用于从表中删除与查询结果集中的行相对应的记录。
- 追加查询: 把查询结果添加到另一个表, 但要注意两者之间的格式与类型要相同。
- 更新查询: 根据查询结果集中的行改变表中现有记录的相应字段的值。

1. 创建生成表查询

【例 5.14】 使用生成表查询, 将学生选课成绩在 60 分以下的学生信息保存到一个新表中。

主要操作步骤如下:

- (1) 打开查询设计视图, 将“学生”、“课程”、“成绩”表添加到窗口中。
- (2) 单击“查询类型”按钮或选择“查询”菜单中的“生成表查询”命令, 打开如图 5.25 所示的“生成表”对话框。

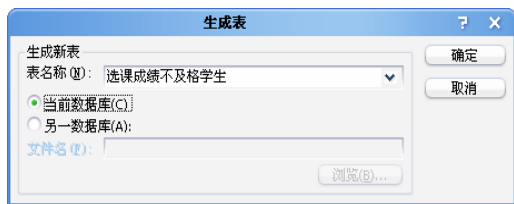


图 5.25 “生成表”对话框

在对话框中, 输入新表的名称, 并选择保存到当前数据库还是另一数据库中。

(3) 在查询设计视图中设置所需的各个选项, 与选择查询设计相同, 设计结果如图 5.26 所示。

(4) 保存所作的设置, 单击“运行”按钮, 系统会弹出“创建新表”对话框, 通过选择“是/否”向新表粘贴记录, 确定是否创建新表。

也可以先保存查询, 以后再运行查询来生成表。若运行了生成表查询, 在数据库的表对象中, 可以查看所生成的表。

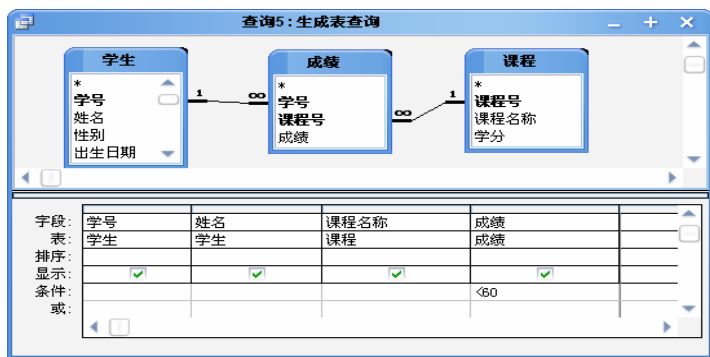


图 5.26 生成表查询设计视图

2. 创建删除查询

删除查询可以从表中删除符合条件的记录，且所作的删除操作无法撤销。删除查询可以从单个表中删除记录，也可以从多个相互关联的表中删除记录。如果要从多个表中删除相关记录，多个表之间必须满足以下条件：

- 在“关系”窗口定义相关表之间的关系。
- 在“关系”窗口中选中“实施参照完整性”复选框。
- 在“关系”窗口中选中“级联删除相关记录”复选框。

【例 5.15】在“学生”、“课程”和“成绩”表中，已经将学生选课成绩在 60 分以下的学生信息保存到一个新表中，使用删除查询从成绩表中删除选课成绩在 60 分以下的学生记录。

主要操作步骤如下：

- (1) 打开查询设计视图，将“成绩”表添加到窗口中。
- (2) 单击“查询类型”按钮，从下拉菜单中选择“删除查询”命令。这时，查询设计视图中添加了一个“删除”行。
- (3) 单击“成绩”字段列表中的“*”号，并将其拖到“字段”行的第 1 列上，第一列将显示“成绩.*”，同时，在该字段“删除”单元格中显示“From”，表示从何处删除记录。
- (4) 将“成绩”字段拖到“字段”行的第 2 列上，同时，在该字段“删除”单元格中显示“Where”，表示要删除哪些记录。在该字段的“条件”单元格中输入条件：<60。设计视图结果如图 5.27 所示。

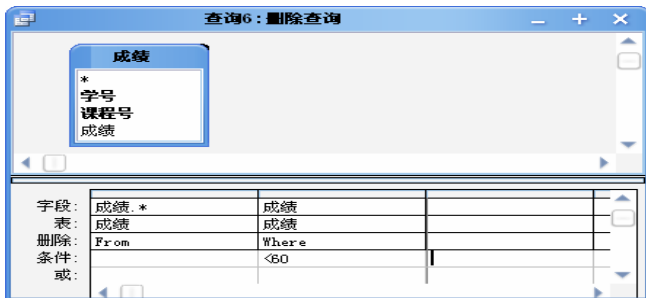


图 5.27 删除查询设计视图

(5) 单击工具栏上的“视图”按钮,可以先预览要删除的记录,若不是要删除的记录,可以返回查询设计视图,修改查询条件;若是要删除的记录,可以单击工具栏上的“运行”按钮,这时系统打开“确认删除”对话框,单击“是”按钮,将从成绩表中永久删除查询到的记录。

【例 5.16】 创建删除查询,删除指定年级学生的记录。

主要设计步骤如下:

(1) 打开查询设计视图,将“学生”表添加到窗口中。

(2) 单击“查询类型”按钮,从下拉菜单中选择“删除查询”命令。这时,查询设计视图中添加了一个“删除”行。

(3) 双击“学生”字段列表中的“*”字段,“学生.*”被添加到“字段”行上,同时,在该列“删除”单元格中显示“From”;在“字段”行的第二列中输入:Mid([学号],4,2),在该列的“条件”单元格中输入:[输入要删除的班级:]。

设计视图结果如图 5.28 所示。

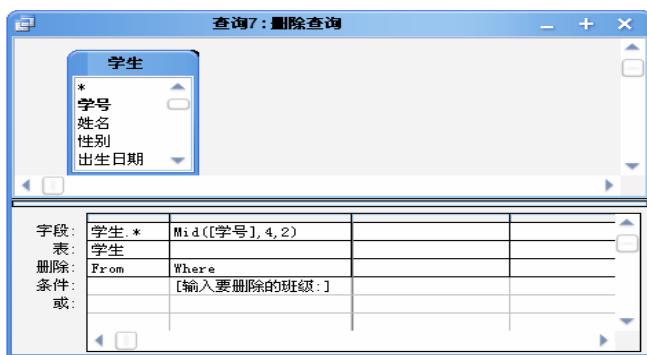


图 5.28 删除查询设计视图

比较例 5.15 和例 5.16 的执行结果,用户可以加深认识选择“级联删除相关记录”的含义。在第一个例子中,仅把“成绩”表中选课成绩在 60 分以下的记录删除,不影响“学生”表的数据记录,而第二个例子中,从“学生”表中删除某个班级的学生记录,则会同时删除“成绩”表中这些学生的选课记录。

3. 创建追加查询

生成表查询用查询选择的字段创建新表结构,然后写入查询记录。而追加查询是对原数据库表进行追加记录的操作,它提供了一个不用到表中就可以增加记录的方法。

【例 5.17】 假定在“学生管理”数据库中,建有一个“大型数据库技术”成绩表,存放着本学期该门课的考试成绩,其中含有:学号、姓名和分数字段。现在,要利用追加查询将其数据记录追加到“成绩”表中。

主要操作步骤如下:

(1) 打开查询设计视图,将“大型数据库技术”表添加到窗口中。

(2) 单击“查询类型”按钮,从下拉菜单中选择“追加查询”命令。系统会弹出“追加”对话框,如图 5.29 所示。在“表名称”中输入要追加记录的表:成绩,单击“确定”

按钮, 这时, 查询设计视图中就添加了一个“追加到”行。

(3) 设置需要追加到目标表的各个选项, 在“追加到”一栏中选择目标表对应的字段, 设计视图结果如图 5.30 所示。

(4) 单击“视图”按钮, 可以查看要追加的记录。确认无误后, 单击“运行”按钮, 系统会弹出追加记录提示框, 单击“是”按钮, 将选择记录追加到目标表中。



图 5.29 “追加”对话框

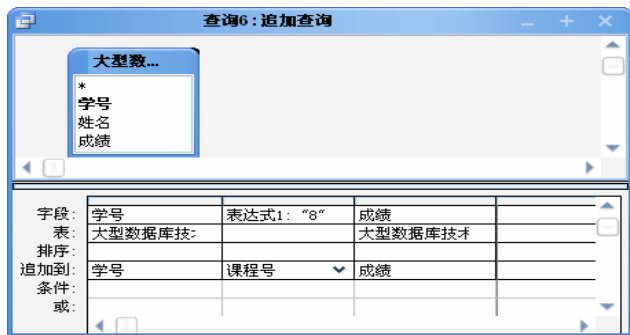


图 5.30 追加查询设计视图

需要说明的是: 由于在“大型数据库技术”表中没有“课程号”字段, 因此可以使用表达式“8”(该门课的课程号)为“成绩”表中的“课程号”字段赋值。

一般情况下, 把追加表的字段拖到“字段”行后, 系统会自动在“追加到”行给出相对应的字段, 也可以选择要追加到的对应字段。

4. 创建更新查询

更新查询可以改变表中记录的数据值。当需要用相同表达式更新许多记录的字段值时, 这种查询是很有用的。例如, 需要按固定的比率增加或者降低所有产品或某一类特定产品的单位价格等。

要更新的记录就是查询中通过设置条件所查询到的记录。

【例 5.18】 在成绩表中, 把所有选课成绩不及格的“成绩”字段值置为 0。

主要设计步骤如下:

(1) 打开查询设计视图, 将“成绩”表添加到窗口中。

(2) 单击“查询类型”按钮, 从下拉菜单中选择“更新查询”命令, 这时, 查询设计视图中添加了一个“更新到”行。

(3) 单击“成绩”字段列表中的“成绩”字段, 并将其拖到“字段”行的第 1 列上,

在“更新到”一行输入数值：0，在“条件”行输入更新条件：<60。设计视图结果如图 5.31 所示。

(4) 单击“视图”按钮，可以查看要更新的记录。确认无误后，单击执行查询按钮，系统会弹出更新提示框，单击“是”按钮，将完成对成绩表中选课成绩不及格记录的“成绩”字段更新。

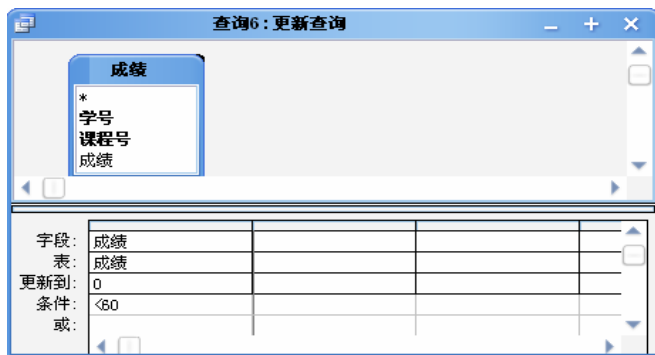


图 5.31 更新查询设计视图

5.4.5 重复项、不匹配项查询

1. 创建查找重复项查询

查询数据表中字段值重复的记录。在数据表中，除设置为主键的字段不能重复外，其他字段允许有重复值。

在 Access 2003 中，可能需要对数据表中某些具有相同值的记录进行检索、分类。利用系统提供的“查找重复项查询向导”可以快速完成一个查找重复项查询。

创建查找重复项查询的步骤如下：

(1) 在“新建查询”对话框中，选择“查找重复项查询向导”，并在向导对话框中选择查询所基于的表或查询，例如，选择“成绩”表，以查询每门课被学生选修的情况。

(2) 选取设为重复值的字段，系统会按照选取的字段自动对数据表中的记录进行检索，如选择“课程号”。

(3) 可以选择另外查询的字段。

(4) 为查询指定一个名字后，单击“完成”按钮，即完成查找重复项查询的建立。

2. 建立不匹配查询

不匹配查询就是在一个表中搜索另一个表中没有相关记录的记录行。在 Access 2003 中，可能需要对数据表中的记录进行检索，查看它们是否与其他表的记录相关，是否真正具有实际意义。

建立不匹配查询的步骤如下：

(1) 在“新建查询”对话框中，选择“查找不匹配项查询向导”，并在向导对话框中选择查询所基于的表或查询，例如，选择“课程”表，以查询没有被学生选修的课程有哪些。

- (2) 选择不包含匹配记录的表,在此,选择“成绩”表。
- (3) 选择在两个表中的匹配字段,在此选择“课程”和“成绩”表的“课程号”字段。
- (4) 为查询指定一个名字后,单击“完成”按钮,即完成查找不匹配项查询的建立。

5.5 SQL 查询

SQL 是结构化查询语言(Structured Query Language)的缩写。SQL 包括数据定义、数据查询、数据操纵和数据控制 4 个部分,是一种功能齐全的数据库语言。该语言于 20 世纪 70 年代就由 Boyce 和 Chamberlin 提出,在 IBM 大型计算机上实现(当时称为 SEQUEL 语言)。由于 SQL 具有语言简洁、方便实用、功能齐全等突出优点,很快得到推广和应用。

SQL 有两种使用方法,一种是以与用户交互的方式联机使用,另一种是作为子语言嵌入到其他程序设计语言中使用。

5.5.1 SQL 的数据定义

SQL 的数据定义功能是指定义数据库的结构,包括定义基本表、定义视图和定义索引 3 个部分。由于视图是基于基本表的虚表,索引是依附于基本表的,因此 SQL 通常不提供修改视图定义和修改索引定义的操作。如果想修改视图定义或索引定义,只能先将它们删掉,然后再重建,不过有些产品如 Oracle 允许直接修改视图定义。

SQL 数据定义语句如表 5.1 所示。

表 5.1 SQL 的数据定义语句

操作对象	操 作 方 式		
	创 建	删 除	修 改
表	CREATE TABLE	DROP TABLE	ALTER TABLE
视图	CREATE VIEW	DROP VIEW	
索引	CREATE INDEX	DROP INDEX	

1. 定义基本表

建立数据库的第一步就是定义基本表。

定义基本表的命令为 CREATE TABLE, 其格式如下:

```
CREATE TABLE <表名> (<列名 1><数据类型>[列级完整性约束条件]
                        [, <列名 2><数据类型>[列级完整性约束条件].....]
                        [, <表级完整性约束条件>]);
```

其中, <表名>是所要定义的基本表的名字,它可以由一个或多个属性(列)组成。建表的同时通常还可以定义与该表有关的完整性约束条件,这些完整性约束条件被存入系统的数据字典中,当用户操作表中的数据时,由 DBMS 自动检查该操作是否违背这些完整性约束条件。如果完整性约束条件涉及该表的多个属性列,则必须定义在表级上,否则既可以

定义在列级，也可以定义在表级。

【例 5.19】 建立一个“学生”表，它由学号、姓名、性别、年龄和所在系 5 个属性组成，其中学号属性不能为空，并且其值是唯一的。

```
CREATE TABLE 学生(学号 CHAR(5) NOT NULL UNIQUE,  
                  姓名 CHAR(8),  
                  性别 CHAR(2),  
                  年龄 INT,  
                  所在系 CHAR(15));
```

其中 NOT NULL 表示该列不能为空值，UNIQUE 表示该属性值唯一。

2. 修改基本表

由于应用环境和应用需求的变化，有时可能需要修改已建立好的基本表，包括增加新列、增加新的完整性约束条件、修改原有的列定义或删除已有的完整性约束条件等。

修改基本表的命令为 ALTER TABLE，其格式为：

```
ALTER TABLE <表名>[ADD<新列名><数据类型>[完整性约束]]  
                  [DROP<完整性约束>]  
                  [MODIFY<列名><数据类型>];
```

【例 5.20】 在学生表中增加“入学时间”列，设置其数据类型为日期型。

```
ALTER TABLE 学生 ADD 入学时间 DATE;
```

【例 5.21】 将学生表中的年龄的数据类型改为半字长整数。

```
ALTER TABLE 学生 MODIFY 年龄 SMALLINT;
```

【例 5.22】 删除学生表中学号必须取值唯一的约束。

```
ALTER TABLE 学生 DROP UNIQUE(学号);
```

SQL 没有提供删除属性列的语句，用户只能间接实现这一功能，即先将原表中要保留的列及其内容复制到一个新表中，然后删除原表，并将新表重命名为原表名。

3. 删除基本表

可用 DROP TABLE 命令删除基本表，其格式为：

```
DROP TABLE <表名>;
```

【例 5.23】 删除学生表。

```
DROP TABLE 学生;
```

基本表一旦删除，表中的数据和在此表上建立的索引都将自动被删掉，而建立在此表上的视图虽然仍然保留，但已无法引用。

5.5.2 SQL的数据操纵

数据操纵是指对表中的数据进行查询、插入、删除和更新等操作。

1. SQL查询

SELECT 查询是数据库的核心操作。

其命令格式为：

```
select [ALL | DISTINCT] <目标列表表达式 1>[,<目标列表表达式 2>].....  
from <表名或视图名列表>  
[where <条件表达式>]  
[group by <分组属性名> [having<组选择条件表达式>]]  
[order by <排序属性名> ] [asc|dEsc];
```

SELECT 语句的含义是，从指定的表或视图中找出符合条件的记录，按目标列表表达式的设定，选出记录中的字段值形成查询结果。

说明：<目标列表表达式>：要查询的数据，一般是列名或表达式。

FROM 子句：数据来源，即从哪些表或视图中查询。

WHERE 子句：查询条件，即选择满足条件的记录。

GROUP BY 子句：对查询结果进行分组。

HAVING 子句：限定分组的条件，必须在 GROUP BY 子句后用。

ORDER BY 子句：对查询结果进行排序，ASC 表示升序，DESC 表示降序。

我们以一个简单的图书管理关系数据模型为基础，通过示例来介绍 SELECT 语句的使用方法。

设图书管理关系数据模型包括以下 3 个关系模式：

图书(总编号，分类号，书名，作者，出版单位，职称，地址)。主码为总编号。

读者(借书证号，姓名，性别，单位，职称，地址)。主码为借书证号。

借阅(借书证号，总编号，借阅日期，备注)。主码为(借书证号，总编号，借阅日期)，外码为借书证号和总编号。

(1) 简单查询

【例 5.24】 找出读者李欣所在的单位。

```
SELECT 姓名,单位  
FROM 读者  
WHERE 姓名= "李欣";
```

【例 5.25】 查看所有读者的全部情况。

```
SELECT *  
FROM 读者;
```

其中 SELECT 子句中的“*”表示所有属性。

【例 5.26】 列出馆中所有藏书的书名及出版单位。

```
SELECT DISTINCT 书名,出版单位  
FROM 图书;
```

DISTINCT 选项 用于从查询结果中去掉重复元组。系统默认为 ALL，即无论重复与否全部给出。

【例 5.27】 查找清华大学出版社的所有图书及单价，结果按单价降序排列。

```
SELECT 书名, 出版单位, 单价
FROM 图书
WHERE 出版单位= "清华大学出版社"
ORDER BY 单价 DESC;
```

【例 5.28】 查找价格在 10 元到 15 元之间的图书，结果按分类号和单价升序排列。

```
SELECT 书名, 作者, 单价, 分类号
FROM 图书
WHERE 单价 BETWEEN 10 AND 15
ORDER BY 分类号, 单价 ASC;
```

【例 5.29】 查找清华大学出版社和科学出版社的所有图书及作者。

```
SELECT 书名, 作者, 出版单位
FROM 图书
WHERE 出版单位 IN( "清华大学出版社", "科学出版社" );
```

【例 5.30】 查找书名以“数据库”打头的所有图书及作者。

```
SELECT 书名, 作者
FROM 图书
WHERE 书名 LIKE "数据库%";
```

(2) 联接查询

简单查询只涉及一个关系，如果查询涉及两个或几个关系，往往要进行联接运算。由于 SQL 是高度非过程化的，用户只要在 FROM 子句中指出关系名称，在 WHERE 子句中写明联接条件即可，联接运算由系统完成并实现优化。

【例 5.31】 查找所有借阅了图书的读者的姓名及所在单位。

```
SELECT DISTINCT 姓名, 单位
FROM 读者, 借阅
WHERE 读者.借书证号=借阅.借书证号;
```

必须注意，如果不同关系中有相同的属性名，为了避免混淆，应当在前面冠以关系名并用“.”分开。

【例 5.32】 查找李晶借的所有图书的书名及借阅日期。

```
SELECT 姓名, 书名, 借阅日期
FROM 图书, 借阅, 读者
WHERE 读者.借书证号=借阅.借书证号
      AND 借阅.总编号=图书.总编号
      AND 姓名="李晶";
```

查询涉及 3 个关系之间的自然联接，用户只需用外码指出联接条件。

(3) 嵌套查询

嵌套查询是指在 SELECT-FROM-WHERE 查询块内部再嵌入另一个查询块，称为子查询，并允许多层嵌套。由于 ORDER 子句是对最终查询结果的表示顺序提出要求，因此它不能出现在子查询中。

【例 5.33】 找出借阅了“C 语言程序设计”一书的读者的姓名及所在单位。

此查询可以用联接查询来完成:

```
SELECT 姓名,单位
FROM 读者,借阅,图书
WHERE 读者.借书证号=借阅.借书证号
      AND 借阅.总编号=图书.总编号
      AND 书名="C 语言程序设计";
```

对于非专业用户来讲, **WHERE** 子句中的条件可能过于复杂, 往往容易丢掉一部分联接条件。下面的嵌套查询形式则清晰、自然, 并可体现出结构化程序设计的优点:

```
SELECT 姓名,单位
FROM 读者
WHERE 借书证号 IN
      (SELECT 借书证号
       FROM 借阅
       WHERE 总编号 IN
            (SELECT 总编号
             FROM 图书
             WHERE 书名="C 语言程序设计"));
```

在执行嵌套查询时, 先对内层子查询求结果, 外层使用内层的查询结果。从形式上看是自下向上进行处理的, 从这个规律出发, 按照手工查询的思路来组织嵌套查询就会显得很简单。

在嵌套查询中最常用的是 **IN**。由于外层查询使用内层查询的结果, 用户事先并不知道内层结果, 因此这里的 **IN** 就不能用一系列 **OR** 来代替。另处, 并非所有的嵌套查询都能用联接查询替代, 有时结合使用更简洁、方便。

【例 5.34】 找出与王鸣一在同一天借了书的读者的姓名、所在单位。

```
SELECT 姓名,单位,借阅日期
FROM 读者,借阅
WHERE 借阅.借书证号=读者.借书证号 AND 借阅日期 IN
      (SELECT 借书日期
       FROM 借阅,读者
       WHERE 借阅.借书证号=借书证号 AND 姓名="王鸣一");
```

(4) 使用聚函数查询

SQL 提供的常用统计函数称为聚函数。这些聚函数使检索功能进一步增强, 它们的自变量是表达式的值, 是按列计算的, 最简单的表达式就是字段名。

SQL 的聚函数有:

- **COUNT**: 计算元组的个数。
- **SUM**: 对某一列的值求和(属性必须是数值类型)。
- **AVG**: 对某一列的值求平均值(属性必须是数值类型)。
- **MAX**: 找出一列值中的最大值。
- **MIN**: 找出一列值中的最小值。

【例 5.35】 计算藏书总册数。

```
SELECT COUNT(*) AS 藏书总册数
FROM 图书;
```

【例 5.36】 查询清华大学出版社图书的最高价格、最低价格和平均价格。

```
SELECT MAX(单价),MIN(单价),AVG(单价)
FROM 图书
WHERE 出版单位= "清华大学出版社";
```

【例 5.37】 查询各个出版社图书的最高价格、最低价格和平均价格。

```
SELECT 出版单位,MAX(单价),MIN(单价),AVG(单价)
FROM 图书
GROUP BY 出版单位;
```

其中 GROUP BY 的作用是按属性的取值对记录分组，然后对每一组分别使用聚函数。在此例中，有几个出版单位就分几个组，按组分别计算最高价格、最低价格和平均价格。

需要注意的是：如果在 SELECT 子句中出现聚函数，与之并列的其他项目必须也是聚函数或者是 GROUP BY 的对象，否则会出现逻辑错误。

【例 5.38】 找出当前至少借阅了 5 本图书的读者的姓名及其所在单位。

```
SELECT 姓名,单位
FROM 读者
WHERE 借书证号 IN
      (SELECT 借书证号
       FROM 借阅
       GROUP BY 借书证号 HAVING COUNT(*)>=5);
```

例中的 HAVING 子句通常跟随在 GROUP BY 之后，其作用是限定检索条件。条件中必须包含聚函数，否则条件可直接放到 WHERE 子句里。

(5) 集合查询

使用 UNION 可以把多个 SELECT 语句的结果进行并操作。

【例 5.39】 查询计算机系的学生及年龄不大于 20 岁的学生。

```
SELECT 学号,姓名
FROM 学生
WHERE 系别="计算机系"
UNION
SELECT 学号,姓名
FROM 学生
WHERE YEAR( DATE( ) )-YEAR( [出生日期] )<=20;
```

使用 UNION 将多个查询结果合并起来时，系统会自动去掉重复元组。需要注意的是：参加 UNION 操作的各查询结果的列数必须相同；对应的数据类型也必须相同。

2. 插入操作

使用 INSERT 语句可以向基本表中插入数据，通常有两种使用形式，一种是插入一个元组，另一种是插入子查询结果，后者可以一次插入多个元组。

(1) 插入单个元组

语句格式为：

```
INSERT INTO <表名>[(<属性 1>,<属性 2>,...)]
```

VALUES (<常量 1>,<常量 2>,...)

其功能是将新元组插入指定表中,其中新元组属性 1 的值为常量 1,属性 2 的值为常量 2,以此类推。

如果某些属性列在 INTO 子句中没有出现,则新元组在这些列上将取空值。

如果 INTO 子句中没有指明任何列名,则新插入的元组必须在每个属性列上均有值。

但必须注意的是:在表定义时说明了 NOT NULL 的属性列不能取空值。

【例 5.40】 向图书(总编号,分类号,书名,作者,出版单位,单价)表中新增一个元组。

```
INSERT INTO 图书
VALUES ("446943", "TP31/138", "微机原理", "杨大华", "清华大学出版社", 17.10);
```

【例 5.41】 向图书表中插入一个元组的部分属性。

```
INSERT INTO 图书(总编号,书名,单价)
VALUES ("44698088", "数据库原理及应用", 16.50);
```

(2) 插入子查询结果

语句格式为:

```
INSERT INTO <表名>[(<属性 1>,<属性 2>,...)]
子查询;
```

其功能是批量插入,一次将子查询的结果全部插入指定表中。

【例 5.42】 建立一个各单位借阅图书情况的统计表,名称为 DWJS,每隔一段时间,如一个月,向此表里追加一次数据。

```
CREATE TABLE DWJS (单位 CHAR(20),
                    借书人数 SMALLINT,
                    借书人次 SMALLINT);
INSERT INTO DWJS (单位,借书人数,借书人次)
SELECT 单位,COUNT(DISTINCT 借书证号),COUNT(总编号)
FROM 借阅,读者
WHERE 读者.借书证号=借阅.借书证号
GROUP BY 单位;
```

3. 删除记录

使用 DELETE 命令可以从表中删除一个或多个元组。

删除元组的格式为:

```
DELETE
FROM <表名>
[WHERE <条件>]
```

【例 5.43】 删除借书证号“9011100”所借总编号为“44698080”的借阅登记。

```
DELETE
FROM 借阅
WHERE 借书证号= "9011100 " AND 总编号= "44698080 ";
```

【例 5.44】 删除借书证号以 90 开头的读者登记和借阅登记。

```
DELETE
FROM 读者
WHERE 借书证号= "90% ";
DELETE
FROM 借阅
WHERE 借书证号= "90% ";
```

4. 更新操作

更新操作又称修改操作。在更新命令中可以用 **WHERE** 子句限定条件，对满足条件的元组给以更新，若不写条件，则对所有元组更新。

更新命令的格式为：

```
UPDATE<表名>
SET <属性名 1>=<表达式 1>[,<属性名 2>=<表达式 2>...]
[WHERE <条件>];
```

其功能是修改指定表中满足条件的元组，用表达式 1 的值替代属性名 1 的值，用表达式 2 的值替代属性名 2 的值，以此类推。

【例 5.45】 将总编号为“44698088”的图书填上作者和出版单位。

```
UPDATE 图书
SET 作者= "王为民", 出版单位= "电子工业出版社"
WHERE 总编号= " 44698088 ";
```

【例 5.46】 将所有图书的单价上调 5%。

```
UPDATE 图书
SET 单价=单价*1.05;
```

5.5.3 SQL视图

任何类型的查询都可以在 SQL 视图中打开，通过修改查询的 SQL 语句，就可以对现有的查询进行修改使之满足用户的要求。

例如，将已建立的“计算机科学与技术专业或者 2004 年入学的学生”查询的条件改为“计算机应用与维护专业或者 2005 年入学的学生”。

主要操作步骤如下：

- (1) 在“设计”视图中打开已建立的查询，结果如图 5.32 所示。
- (2) 选择“视图”菜单下的“SQL 视图”命令，或单击工具栏上“视图”按钮的下三角按钮，从下拉列表中选择“SQL 视图”，这时系统打开如图 5.33 所示的 SQL 视图窗口。
- (3) 在 SQL 视图窗口中，可以直接修改 **WHERE** 子句后面的条件，例如，把专业条件中的“计算机科学与技术”改为“计算机应用与维护”；把入学日期条件中的年度值“2004”改为“2005”，结果如图 5.34 所示。
- (4) 单击工具栏上的“视图”按钮，可以预览查询的结果。单击“保存”按钮，可以保存此次通过 SQL 视图对查询所作的修改。

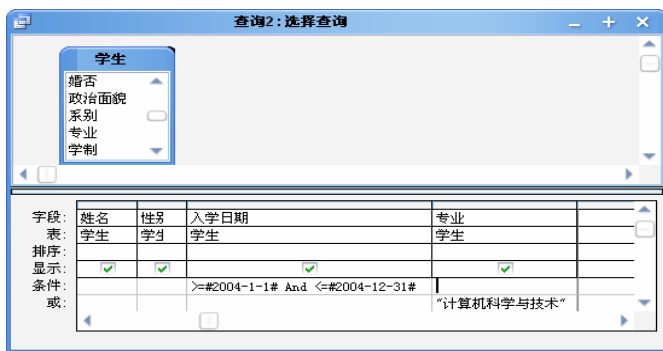


图 5.32 查询设计视图

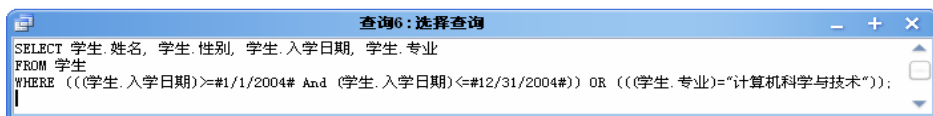


图 5.33 SQL视图

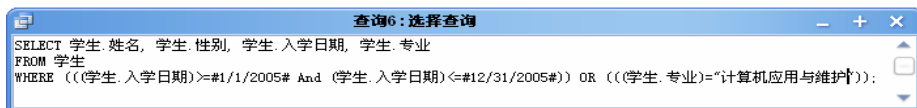


图 5.34 修改后的SQL视图

切换和对比查询视图的设计视图和 SQL 视图，可以帮助用户掌握 SQL 语句的使用，反过来，也可以提高用户在设计视图中建立查询条件的能力。

习 题

1. 查询的功能是什么？查询与表有何不同？
2. 在 Access 2003 中，查询分为几种类型？各具有什么功能？
3. 什么是操作查询？分为几类？
4. 什么是参数查询？其特点是什么？
5. 简述利用设计视图创建选择查询的步骤。
6. 简述利用设计视图创建交叉表查询的步骤。
7. 简述利用设计视图创建参数查询的步骤。
8. 简述创建生成表查询和删除查询的步骤。
9. 使用 SQL 语句创建新表。
10. 使用 SQL 语句修改新表结构。
11. 使用 SQL 语句向新表中插入数据记录。
12. 使用 SQL 语句删除表中已有的数据记录。
13. 使用 SQL 语句修改表中已有的数据。
14. 使用 SQL 语句创建基于多表的查询。

第 6 章 窗体设计及高级应用

本章学习目标

窗体作为人机交互的一个重要接口，是 Access 2003 数据库中功能最强的对象之一，数据的使用与维护大多都是通过窗体来完成的。本章主要介绍窗体的基本知识，包括窗体的基本概念、使用向导创建窗体、使用设计器创建窗体、弹出式窗体以及控件工具箱的使用等内容。通过本章的学习，读者应该掌握以下内容：

- Access 2003 中窗体的构成与作用
- 利用向导创建窗体
- 在设计视图中如何设计窗体
- 窗体中控件对象的使用
- 窗体及控件的属性设置与事件的设计方法

6.1 窗体基础知识

窗体是 Access 2003 数据库中的一个非常重要的对象，同时也是最复杂和灵活的对象。通过窗体用户可以方便地输入数据、编辑数据、显示统计和查询数据，是人机交互的窗口。窗体的设计最能展示设计者的能力与个性，好的窗体结构能使用户方便地进行数据库操作。此外，利用窗体可以将整个应用程序组织起来，控制程序流程，形成一个完整的应用系统。

在 Access 2003 中，窗体具有可视化的设计风格，由于使用了数据库引擎机制，可将数据表捆绑于窗体。

6.1.1 窗体的概念与作用

窗体是在可视化程序设计中经常提及的概念，实际上窗体就是程序运行时的 Windows 窗口，在应用系统设计时称为窗体。

窗体的作用是提供给用户进行操作，是用户与 Access 2003 应用程序之间的主要操作接口，开发数据库应用系统，就必须制作窗体。

对用户而言，窗体是操作应用系统的界面，靠菜单或按钮提示用户进行业务流程操作，不论数据处理系统的业务性质如何不同，必定有一个主窗体，提供系统的各种功能，用户通过选择不同操作进入下一步操作的界面，完成操作后返回主窗体。

窗体的主要特点与作用：

1. 显示与编辑数据

可以通过窗体录入、修改、删除数据表中的数据，该功能是窗体最普遍的应用。例如，图 6.1 所示的“学生信息处理”窗体。

学生信息处理

学生基本信息处理

学号	30203301	政治面貌	团员
姓名	杜少华	系别	计算机系
性别	男	专业	计算机通信工程
出生日期	1985-10-10	学制	4
民族	汉族	年级	2003
身份证号码	37070619851010211X	班级	1
婚否	☒ 已婚 ☐ 未婚	入学日期	2003-9-10
籍贯	山东省济南市		
现住址	山东省济南市		
备注			

添加记录 删除记录 保存记录 撤消记录 打印记录 关闭窗口

首记录 上一条 下一条 末记录

图 6.1 “学生信息处理”窗体

2. 使用窗体查询或统计数据库中的数据

可以通过窗体输入数据查询或统计条件，查询或统计数据库中的数据。该功能也是窗体最普遍的应用。例如，图 6.2 所示的“学生信息查询”窗体。

学生信息查询

学生基本信息查询

选择查询项目: 姓名

输入查询内容: 杜少华

查询 退出

学号	姓名	性别	出生日期	民族
30203301	杜少华	男	1985-10-10	汉族

图 6.2 “学生信息查询”窗体

3. 显示提示信息

用于显示提示、说明、错误、警告等信息，帮助用户进行操作。

6.1.2 窗体构成

窗体通常由窗体页眉、窗体页脚、页面页眉、页面页脚和主体 5 部分组成，每一部分称为窗体的“节”，除主体节外，其他节可通过设置确定有无，但所有窗体必有主体节，其结构如图 6.3 所示。



图 6.3 窗体的结构

窗体页眉：位于窗体的顶部位置，一般用于显示窗体标题、窗体使用说明或放置窗体任务按钮等。

页面页眉：只显示在应用于打印的窗体上，用于设置窗体在打印时的页头信息，例如，标题、图像、列标题、用户要在每一打印页上方显示的内容。

主体：是窗体的主要部分，绝大多数的控件及信息都出现在主体节中，通常用来显示记录数据，是数据库系统数据处理的主要工作界面。

页面页脚：用于设置窗体在打印时的页脚信息，例如，日期、页码、用户要在每一打印页下方显示的内容。由于窗体设计主要应用于系统与用户的交互接口，通常在窗体设计时很少考虑页面页眉和页面页脚的设计。

窗体页脚：功能与窗体页眉基本相同，位于窗体底部，一般用于显示对记录的操作说明、设置命令按钮。

需要说明的是：窗体在结构上由以上五部分组成，在设计时主要使用标签、文本框、组合框、列表框、命令按钮、复选框、切换与选项按钮、选项卡、图像等控件对象，以设计出面向不同应用与功能的窗体。

6.1.3 窗体类型

在 Access 2003 数据处理窗体的设计中，根据数据记录的显示方式提供了 6 种类型的窗体，分别是纵栏式窗体、表格式窗体、数据表窗体、图表窗体、数据透视表窗体以及主/子窗体。纵栏式窗体、表格式窗体、数据表窗体是对相同的数据的不同显示形式，其中纵栏式窗体同时只显示一条记录，而表格式窗体和数据表窗体可同时显示多条记录。在新建窗体对话框中，可以看到这些窗体类型，如图 6.4 所示。

图表窗体就是利用 Microsoft Office 提供的 Microsoft Graph 程序以图表方式显示用户的数据，这样在比较数据方面显得更直观方便。在 Microsoft Access 2003 中，用户既可以单独使用图表窗体，也可以在窗体中插入图表控件。

数据透视表窗体是一种交互式的表，可以进行选定的计算，它是 Access 2003 在指定表或查询基础上产生一个导入 Excel 的分析表格，允许对表格中的数据进行一些扩展和其他

的操作。

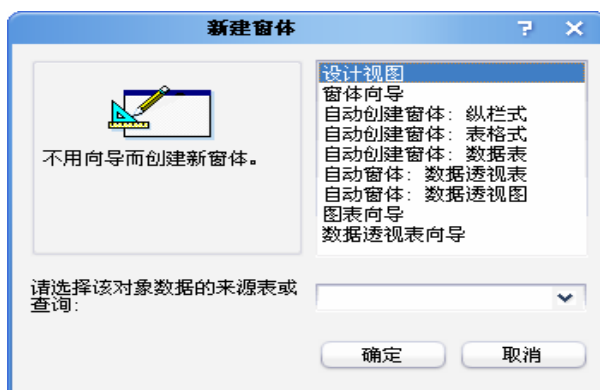


图 6.4 “新建窗体”对话框

主/子窗体也称为阶层式窗体、主窗体/细节窗体或父窗体/子窗体，在显示具有一对多关系的表或查询中的数据时，子窗体特别有效。

另外，依据窗体的其他性质也可对窗体作出另类划分：

- 根据窗体是否与数据源连接可以分为绑定窗体和未绑定窗体，绑定窗体与数据源连接，未绑定窗体不与数据源连接。
- 根据窗体的功能用途，可以将窗体分为几种窗体类型，用于控制程序流程的窗体、用于与用户进行交互的窗体以及用于操作数据库的窗体(上述 6 种)。

控制程序流程的窗体最典型的例子就是切换面板，该面板对浏览数据库很有帮助。切换面板中有一些按钮，单击这些按钮可以打开相应的窗体和报表(或打开其他窗体和报表的切换面板)、退出 Microsoft Access 2003 或自定义切换面板。

用于与用户交互的窗体用来向用户提供系统的信息，也接受用户输入的信息到系统中。一般用户设计的弹出式窗体就是这种用途，另外通过调用系统函数 MsgBox 和 InputBox 也可以实现信息的输入输出。

6.1.4 窗体视图

窗体视图是窗体在具有不同功能和应用范围下呈现的外观表现形式。表和查询有两种视图：设计视图和数据表视图；窗体有 3 种视图：设计视图、窗体视图和数据表视图。

设计视图是创建窗体或修改窗体的窗口，任何类型的窗体均可以通过设计视图来完成创建。在窗体的设计视图中，可直观地显示窗体的最终运行格式，设计者可利用控件工具箱向窗体添加各种控件，通过设置控件属性、事件代码处理，完成窗体功能设计；通过格式工具栏中的工具完成控件布局等窗体格式设计。在设计视图中创建的窗体，可在窗体视图和数据表视图进行结果查看。

窗体视图就是窗体运行时的显示格式，用于查看在设计视图中所建立窗体的运行结果。在窗体设计过程中，需要不断在两种视图之间进行切换，以完善窗体设计。

数据表视图是以行和列的格式显示表、查询或窗体数据的窗口。在数据表视图中，可

以编辑、添加、修改、查找或删除数据。

6.2 创建窗体

在 Access 2003 中可以使用两种方法创建窗体,一种是系统提供的窗体向导;另一种是手动方式(又称窗体设计器)。利用窗体向导可以简单、快捷地创建窗体,Access 2003 会提示设计者输入有关信息,根据输入信息完成窗体创建。一般情况下,即使是经验丰富的设计人员,仍需先利用窗体向导建立窗体的基本轮廓,然后切换到设计视图完成进一步的设计。使用人工方式创建窗体,需要创建窗体的每一个控件,建立控件与数据源的联系,设置控件的属性等。这种方法可以为用户提供最大的灵活性,完成功能强大、格式美观的窗体,有经验的设计者均是通过设计视图来完成窗体设计的。

Access 2003 提供了 6 种创建窗体的向导,如图 6.4 所示,包括窗体向导、自动创建窗体:纵栏式、自动创建窗体:表格式、自动创建窗体:数据表、图表向导和数据透视表向导。

6.2.1 使用自动创建窗体向导

如果用户只需要创建一个简单的数据维护窗体,显示选定表或查询中所有字段及记录,可使用自动创建窗体向导。

自动创建窗体有:纵栏式、表格式、数据表 3 种格式,创建过程完全相同。

使用自动创建窗体向导的具体操作步骤如下:

- (1) 在“数据库”窗口中,选择“窗体”对象。
- (2) 单击“数据库”窗口工具栏上的“新建”按钮。
- (3) 在“新建窗体”对话框中,选择下列向导之一。
 - “自动创建窗体:纵栏式”,每个字段都显示在一个独立的行上,并且左边带有一个标签(标题为对应字段名)。
 - “自动创建窗体:表格式”,每个记录的所有字段显示在一行上,标签显示在窗体的顶端。
 - “自动创建窗体:数据表”,每个记录的字段以行和列的格式显示,即每个记录显示为一行,每个字段显示为一列。字段的名称显示在每一列的顶端。
- (4) 窗体数据源选择,在“请选择该对象数据的来源表或查询:”后的下拉列表框中选择作为窗体数据来源的表或查询。
- (5) 单击“确定”按钮,保存窗体,结束窗体的创建。

需要注意的是:Access 2003 应用最近用于窗体的“自动格式”。如果以前没有使用向导创建过窗体或没有使用过“格式”菜单上的“自动套用格式”命令,Access 2003 将使用“标准”格式。

【例 6.1】在“学生管理”数据库中,以“学生”表作为数据源,使用“自动创建窗体:纵栏式”创建窗体,结果如图 6.5 所示。

学号	30203301	班级	1	照片
姓名	杜少华	入学日期	2003-9-10	
性别	男	籍贯	山东省济南	
出生日期	1985年10月10日	现住址	山东省济南	
民族	汉族			
身份证号码	370706198510102			
婚否	☐			
政治面貌	团员	备注		
系别	计算机系			
专业	计算机通信工程			
学制	4			
年级	2003			

图 6.5 “自动创建窗体：纵栏式”窗体

6.2.2 使用“窗体向导”

在使用自动创建窗体向导创建窗体时，作为数据源的表或查询中的字段默认方式为全部选中，窗体布局的格式也已确定，如果用户要选择数据源中的字段及窗体的布局和窗体样式，可以使用“窗体向导”来创建窗体。根据数据源的选择，一般把使用“窗体向导”创建窗体分为单数据源和多数据源两种情况。

1. 使用向导创建基于一个表或查询的窗体

基本操作步骤如下：

- (1) 在“新建窗体”对话框中，双击“窗体向导”选项，进入“窗体向导”对话框。
- (2) 在“表/查询”下拉列表框中选择作为窗体数据源的表或查询的名称。
- (3) 在“可用字段”列表框中选择需要在新建窗体中显示的字段。使用“>”按钮逐个添加或使用“>>”按钮全部添加到“选定的字段”列表框。“<”与“<<”反向处理。
- (4) 连续单击“下一步”按钮，选择窗体布局格式及窗体样式。
- (5) 为窗体指定标题。
- (6) 单击“确定”按钮，保存窗体，结束创建。

2. 创建基于多个表的窗体

在“学生管理”数据库中，学生基本信息存放在“学生”表中，选修课成绩存放在“成绩”表中。如何在一个窗体中完成：在浏览学生基本信息的同时，查看该生的选课及成绩？这就需要创建基于多表的窗体。

基于多表的窗体一般称为多表窗体，从多个表或查询中提取数据。常用的方法是建立

主/子窗体，但在创建窗体之前，要确保作为主窗体的数据源与作为子窗体的数据源之间建立了“一对多”的关系。在 Access 2003 中，创建主/子窗体的方法有两种：一是同时创建主窗体与子窗体，二是将已有的窗体作为子窗体添加到另一个已有窗体中。子窗体既可以固定于主窗体之中，也可以是弹出式子窗体。

【例 6.2】在“学生管理”数据库中，以“学生”表和“成绩”表作为数据源，创建如图 6.6 所示的窗体。

图 6.6 主/子窗体

主要操作步骤如下：

(1) 在“新建窗体”对话框中单击“窗体向导”，然后单击“确定”按钮，或双击“窗体向导”，进入“窗体向导”对话框之一，如图 6.7 所示。

图 6.7 “窗体向导”对话框之一

(2) 在图 6.7 对话框中，打开“表/查询”下拉列表，从中选择“表：学生”，在“可用字段”列表中选择要显示的字段，使用“>”或“>>”按钮添加要显示的字段。再打开

“表/查询”下拉列表，从中选择“表：成绩”，在“可用字段”列表中选择要显示的字段，使用“>>”添加要显示的字段。然后单击“下一步”，进入“窗体向导”对话框之二，如图 6.8 所示。

(3) 在图 6.8 对话框中，要求确定窗体查看数据的方式，由于数据来源于两个表，有两个选项：通过“学生”表或“成绩”表查看。由于“学生”表和“成绩”表之间具有一对多关系，“学生”表位于一对多关系中的“一”方，所以应选择“学生”表。

选择“带有子窗体的窗体”单选项。然后单击“下一步”，进入“窗体向导”对话框之三，如图 6.9 所示。

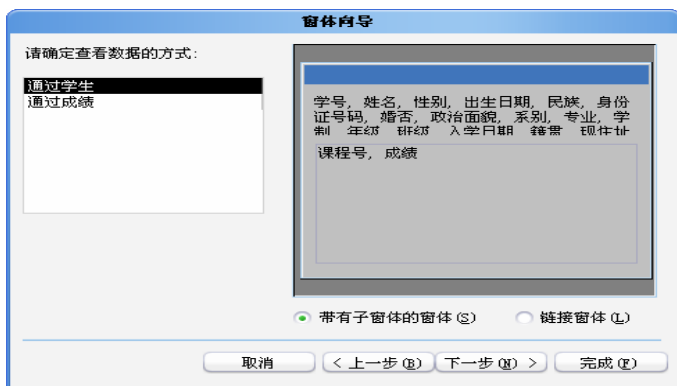


图 6.8 “窗体向导”对话框之二



图 6.9 “窗体向导”对话框之三

(4) 在图 6.9 对话框中，要求确定窗体所采用的布局。有两个可选项：“表格”和“数据表”。选中其中一项，其布局结果在左侧显示，在此选中“数据表”，单击“下一步”，进入“窗体向导”对话框之四，如图 6.10 所示。



图 6.10 “窗体向导”对话框之四

(5) 在图 6.10 对话框中，要求确定窗体所采用的样式。右边列表框中给出了若干种窗体的样式。用户可根据需要选择其中一种，其样式结果在左侧显示，在此选中“标准”，单

击“下一步”，进入“窗体向导”对话框之五，如图 6.11 所示。

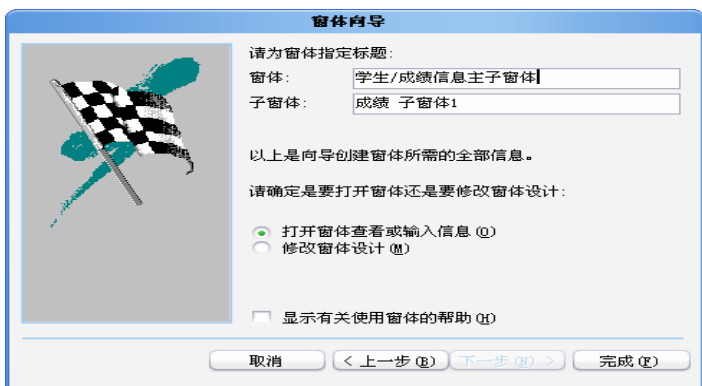


图 6.11 “窗体向导”对话框之五

(6) 在图 6.11 对话框中，要求确定主/子窗体的窗体标题，在“窗体”后文本框中输入主窗体标题“学生/成绩信息主子窗体”。

(7) 单击“完成”按钮，完成如图 6.6 所示的主/子窗体的创建。

需注意的是：如果用户在选择记录源和字段的操作中，选择的多个表之间没有任何关系，单击“下一步”按钮后窗体向导将提示用户重新定义相应的表间关系。

6.2.3 使用“图表向导”

在实际应用中，将表或查询中的数据及其之间的关系用图表形象地加以描述，更能直观地反映数据处理结果。利用 Access 2003 提供的“图表向导”可以快速创建图表窗体，要使用图表窗体，用户需要安装 Microsoft Graph。

使用“图表向导”创建图表窗体，可以按照下例所示步骤进行。

【例 6.3】在“学生管理”数据库中，以建立的“统计各专业学生人数”查询为数据源，使用“图表向导”创建窗体，显示如图 6.12 所示的统计结果。

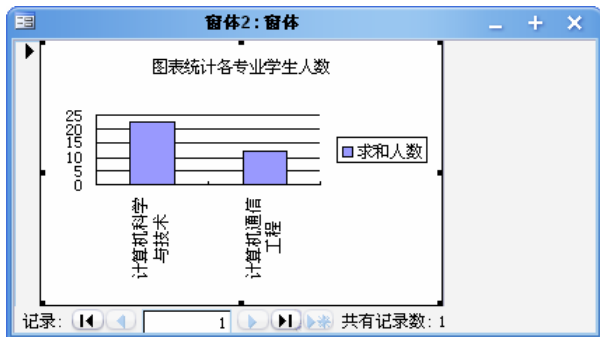


图 6.12 “图表向导”统计各专业学生人数

主要操作步骤如下：

(1) 在“新建窗体”对话框中选择“图表向导”选项，在“请选择该对象数据的来源表或查询”下拉列表框中，选择“统计各专业学生人数”查询作为窗体数据来源。

(2) 单击“确定”按钮启动“图表向导”对话框之一，如图 6.13 所示。

(3) 在图 6.13 对话框中为图表选择所需的字段。在“可用字段”列表框中选择“专业”和“人数”字段，单击“>”按钮添加到“用于图表的字段”列表中。单击“下一步”按钮，出现如图 6.14 所示的“图表向导”对话框之二。

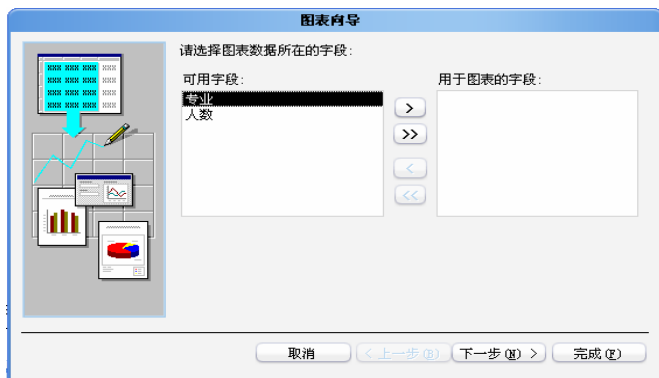


图 6.13 “图表向导”对话框之一



图 6.14 “图表向导”对话框之二

(4) 在图 6.14 中，选择一种图表类型，如选择“三维柱形图”，单击“下一步”按钮，进入“图表向导”对话框之三，如图 6.15 所示。

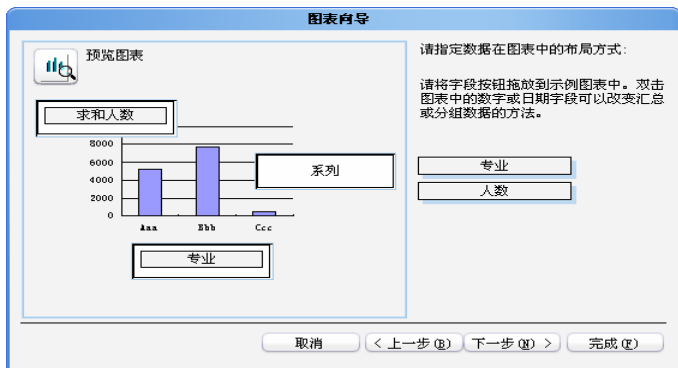


图 6.15 “图表向导”对话框之三

(4) 在图 6.15 中, 需设置各字段在图表中如何显示。此例中, 选“专业”为横坐标, “人数”为纵坐标, 单击“下一步”按钮, 出现如图 6.16 所示的“图表向导”对话框之四。

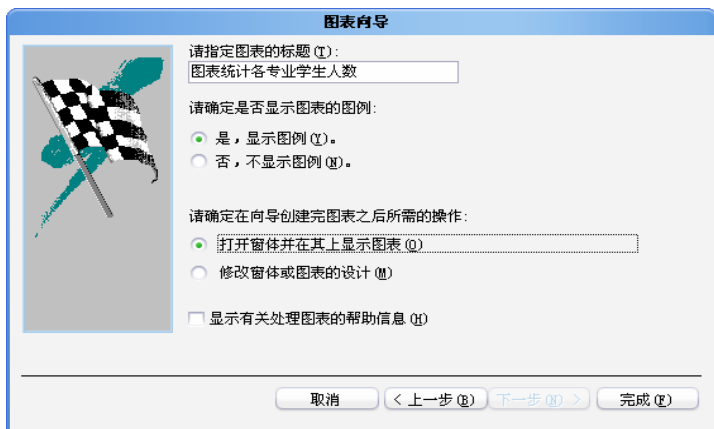


图 6.16 “图表向导”对话框之四

需注意的是: 在图 6.15 中, 坐标图下面为横轴框, 右边为系列框, 上边为纵轴框。向导已将默认字段放置在各框中, 根据需要, 用户可将相应字段拖离各框, 或可将其他字段拖进各框。双击“纵轴框”, “向导”将打开“汇总”对话框, 在此对话框中可以改变对数据的汇总方式。

(5) 在图 6.16 中, 为创建的图表指定标题, 在“请确定是否显示图表的图例”选项组中, 可以指定是否显示图表图例。单击“完成”按钮, 完成如图 6.12 所示图表的创建。

6.2.4 使用“数据透视表向导”

数据透视表窗体是一种交互式的表, 可以进行选定的计算, 例如求和(“数字”字段的默认值)与计数(“文本”字段的默认值), 所进行的计算与数据在数据透视表窗体中的排列有关。例如, 数据透视表窗体可以水平或者垂直显示字段值, 然后计算每一行或列的合计。数据透视表也可以将字段值作为行号或列标, 在每个行列交汇处计算出各自的数量, 然后计算小计和总计。使用数据透视表窗体, 需要用户安装 Microsoft Excel。

【例 6.4】 创建计算各系不同职称教师人数的数据透视表窗体。

基本思想是: 将“专业”作为列标题放在数据透视表的顶端, “系别”作为行标题放在数据透视表的左列, 在行列的交叉处显示计算出来的相应教师人数, 结果如图 6.17 所示。

主要操作步骤如下:

- (1) 在“新建窗体”对话框中, 单击“数据透视表向导”选项, 再单击“确定”按钮。
- (2) 打开“数据透视表向导”对话框之一, 单击“下一步”按钮, 打开“数据透视表向导”对话框之二, 如图 6.18 所示, 提示用户选择表或查询中的字段。
- (3) 在图 6.18 所示窗口中, 选择“编号”、“系别”和“职称”3 个字段, 单击“完成”按钮, 打开数据透视表窗体, 如图 6.19 所示, 此时窗体是空白的。

教师表						—	+	×
将筛选字段拖至此处								
职称 ▼								
	教授	副教授	讲师	助教	总计			
系列 ▼	编号	的计数	编号	的计数	编号	的计数		
计算机	+	4	2	3		9		
经济	+	2	3	2		7		
生物	+	1	2	3	1	7		
数学	+	3	2	1	1	7		
体育	+		4	3		7		
物理	+	4	3			7		
英语	+	3		2	1	6		
总计	+	17	16	14	3	50		

图 6.17 统计各系不同职称人数的数据透视表

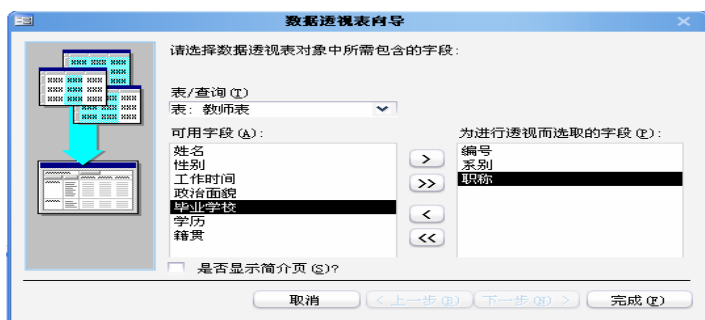


图 6.18 “数据透视表向导”对话框之二



图 6.19 数据透视表窗体

(4) 在图 6.19 所示窗口中，将“系别”拖至行字段，将“职称”拖至列字段，然后“编号”拖至汇总处，得到结果如图 6.17 所示的最终窗体。

6.3 自定义窗体

利用上节介绍的“自动创建窗体”、“窗体向导”和“图表向导”等向导工具可以创建多种窗体，但这只能满足一般的显示与功能要求。由于应用程序的复杂性和功能要求的多样性，使用向导所创建的窗体在实际应用中并不能很好地满足要求，而且有一些类型的窗体用向导无法创建。例如，在窗体中增加各种按钮，实现数据的检索，加入说明性信息，打开、关闭 Access 2003 对象等，因此，Access 2003 提供了窗体设计器，使用窗体设计器，

专业人员可以设计出功能更强大、界面更友好的窗体。

窗体设计器就是窗体的设计视图。在窗体的设计视图中，利用工具箱可以向窗体添加各种控件；利用属性窗口可以设置控件的属性、定义窗体及控件的各种事件过程、修改窗体的外观。窗体设计的核心即是控件对象设计。

本节将介绍窗体设计工具箱的使用、对象属性及设置、对象事件及应用和其常用方法。

6.3.1 窗体设计视图的组成与主要功能

窗体的设计视图主要由窗体设计区域及窗体设计工具栏、控件工具箱、弹出式菜单、格式工具栏等辅助工具组成。各种工具在窗体设计中起不同的作用，用于辅助完成窗体的设计。

1. 窗体设计工具栏

窗体设计工具栏包含有各种命令按钮，这些命令按钮可以在设计窗体时使用。如图 6.20 所示。



图 6.20 窗体设计工具栏

表 6.1 列出“窗体设计”工具栏中各个工具按钮的作用，表中的顺序按照图从左到右排列。

表 6.1 “窗体设计”工具栏中提供的按钮及其作用描述

按钮名称	功 能 描 述
视图	显示当前窗口的可用视图。单击按钮旁边的箭头，选择所需的视图
保存	保存当前正在设计的窗体
搜索	在选定的位置中搜索包含特定文本的文件
打印	立即打印选定的视图，而不显示“打印”对话框
打印预览	显示窗体打印时的效果，可以对页面进行缩放，以便一次显示一页或多页
剪切	移去选定的内容，并放置到剪贴板上，然后可以将其插入到其他地方。如果要还原最近的剪切动作，可选择“编辑”菜单中的“撤销剪切”命令
复制	复制选定的内容到剪贴板上，然后可以将其插入到其他地方
粘贴	将剪贴板中选定的内容插入到活动的数据库对象中。使用“复制”与“粘贴”或“剪切”与“粘贴”可以复制或移动选定的内容，如果要还原粘贴操作，单击“编辑”菜单中的“撤销粘贴”命令
格式刷	从一个控件复制格式(如，颜色、线条样式和字体属性)到另一个控件。单击“格式刷”按钮可以把格式一次复制到一个控件，双击可以把格式一次复制到多个控件。完成后按 Esc 键即可

续表

按钮名称	功 能 描 述
撤销	撤销最近执行的可还原操作, 撤销命令的名称取决于最近一次执行的操作(如“撤销剪切”或“撤销移动”)。如果操作不能撤销, 撤销命令的名称将改为“无法撤销”
恢复	恢复最近执行的撤销操作
插入超级链接	插入或修改一个超级链接地址或统一资源定位符 (URL)
字段列表	显示窗体基础记录源所包含的字段列表。从列表中拖动字段可以创建自动结合到记录源的控件
工具箱	显示或隐藏工具箱。窗体中的控件通过工具箱来创建
自动套用格式	将事先定义的格式应用于窗体
代码	在“模块”窗口中显示选定窗体所包含的代码
属性	显示选定项目的属性表, 如果不选择任何项目, 则显示当前活动对象的属性表
生成器	显示选定项目或属性的生成器。Access 2003 只有在选定项目或属性具有可用的生成器时, 才能激活该按钮
数据库	显示“数据库”窗口, 列出当前数据库中的全部对象。可以利用拖放等方法将对象从“数据库”窗口移到当前窗口
新对象	利用向导创建数据库对象
Office 助手	让“Office 助手”提供帮助主题和提示信息

2. 格式工具栏

除了窗体设计工具栏外, Access 2003 还提供了格式工具栏。利用格式工具栏可设置窗体或其控件的文本格式。格式工具栏如图 6.21 所示。



图 6.21 格式工具栏

格式工具栏中各种工具按钮的功能与 Office 其他组件基本一致, 在此不作介绍。

3. 窗体弹出式菜单

用户在窗体设计窗口中不同的地方右击会有不同的弹出式菜单, 在已放置好的控件上右击会出现控件相关的弹出式菜单。在窗口的标题栏上右击会看到图 6.22 所示的弹出式菜单。

菜单中的最上面四项用于切换窗体的视图, “事件生成器”用于该窗体的事件过程代码设计, 用户可以分别选择生成表达式、宏或代码。“Tab 键次序”用于调整窗体中控件的 Tab 键次序, Tab 键次序指的是在窗体运行视图中用户通过按 Tab 键或 Shift +Tab 键切换窗体中各个控件获得焦点的次序。“页面页眉/页脚”和“窗体页眉/页脚”分别用于切换窗体是否要页面的页眉页脚和窗体的页眉页脚。“属性”命令用于打开属性窗口, 从而对窗体的属性进行设置。



图 6.22 窗体的右键菜单

4. 控件工具箱

控件工具箱提供了用于窗体设计的各种控件对象，利用控件工具箱可以向窗体上添加各种控件。有些控件对象的使用提供了向导使用模式，可以帮助开发者加快窗体的设计过程。通过工具箱及其相应向导，初学者也可以逐渐掌握 Access 2003 窗体的应用。工具箱实际上也是工具栏的一种，与 Access 2003 其他工具栏一样也具有两种状态，即停靠状态和游离状态。工具箱中有 18 种工具按钮用来向窗体中添加控件，此外还有两个按钮分别用于选择控件对象和设置控件向导的有效性。停靠状态的工具箱如图 6.23 所示。



图 6.23 窗体设计的工具箱

(1) 打开和关闭工具箱

选择“视图”菜单中的“工具箱”命令或单击窗体设计工具栏中的“工具箱”按钮，可以显示或隐藏“工具箱”。

(2) 工具箱的移动与锁定

用鼠标指向工具箱的标题栏，按下鼠标左键拖动，可将工具箱移动到不同的目标位置。如果要重复使用工具箱中的某个控件对象，可以锁定该对象，该对象被锁定后，重复使用时不必每次单击该对象。要锁定控件对象，双击要锁定的控件对象即可；要解除锁定，按 Esc 键即可。

(3) 使用工具箱向窗体中添加控件

利用工具箱向窗体中添加各种控件时，首先单击“工具箱”中相应的工具按钮，然后在窗体上单击或拖动，如果所添加的控件具有向导且“控件向导”按钮已按下，Access 2003 将自动启动相应的控件向导，用户可按照向导的提示进行操作以完成控件的添加，控件添加完成后，通过右击相应控件，选择快捷菜单中的“属性”命令来设置控件的属性。工具箱中的各种工具按钮的作用如表 6.2 所示。

表 6.2 工具箱中控件及其功能描述

控件名称	功 能
选择对象	此按钮按下时，可以选择窗体上的各种控件
控件向导	用于打开或关闭控件向导。具有向导的控件有：列表框、组合框、选项组、命令按钮、图表、子窗体。要使用这些控件的向导，必须按下“控件向导”按钮
标签	用来显示说明性文本的控件，如窗体上的标题或指示文字。在创建其他控件时 Access 2003 将自动为其添加附加标签
文本框	用来显示、输入或编辑窗体的基础记录源数据，显示计算结果，或者接受输入的数据
选项组	与复选框、选项按钮或切换按钮搭配使用，用于显示一组可选值，只选择其中一个选项。如用选项按钮指定性别为男或女
切换按钮	此按钮可用于：作为结合到“是/否”字段的独立控件，或作为接收用户在自定义对话框中输入数据的非结合控件，或者作为选项组的一部分；切换按钮只有两种可选状态
选项按钮	此按钮可用于：作为结合到“是/否”字段的独立控件，或作为接收用户在自定义对话框中输入数据的非结合控件，或者作为选项组的一部分；选项按钮只能在多种可选状态中选择一种
复选框	与“选项按钮”作用相同
组合框	该控件结合了文本框和列表框的特性，即在组合框中直接输入文字，或在列表中选择输入项，然后将所做选择添加到所基于的字段中
列表框	显示可滚动的数值选项列表。从列表中选择时将改变所基于的字段值
命令按钮	用来完成各种操作，一般与宏或代码联接，单击时执行相应的宏或代码
图像	用来在窗体中显示静态图片。静态图片不是 OLE 对象，一旦添加到窗体中就无法对其进行编辑
未绑定对象	用来在窗体中显示 OLE 对象，不过此对象与窗体所基于的表或查询无任何联系，其内容并不随着当前记录的改变而改变
绑定对象	用来在窗体中显示 OLE 对象，如“人事管理”数据库中人员的照片可保存在一个单独的字段中，利用该控件显示人员照片，当改变当前记录时，该对象随之更新
分页符	通过插入分页符控件，在打印窗体上开始一个新页
选项卡控件	用来创建多页的选项卡对话框。选项卡控件上可以添加其他类型的控件
子窗体/报表	用来显示来自多表的数据
直线	用来向窗体中添加直线，通过添加直线可突出显示某部分内容
矩形	用于将相关的一组控件或其他对象组织到一起以突出显示
其他控件	用于向窗体中添加 ActiveX 控件

6.3.2 属性、事件与方法

1. 属性

属性是对象特征的描述。每一窗体、报表、节和控件都有各自的属性设置，可以利用

这些属性来更改特定项目的外观和行为。使用属性表、宏或 Visual Basic，可以查看并更改属性。关于宏与 Visual Basic 对属性的操作在以后章节介绍，在此仅介绍属性表。

在窗体设计视图中，每当使用“工具箱”向窗体添加某个控件后，可随时设置该控件的属性，设置方法有多种，常用的设置方法是：右击该控件，在弹出的快捷菜单中，选择“属性”调出该控件的属性设置对话框。例如，在窗体加入文本框后，按以上操作调出文本框控件的属性设置对话框，如图 6.24 所示。

控件属性分为：格式属性、数据属性、事件属性和其他属性，各项属性的含义及主要属性的设置将在 6.3.3 小节中介绍。

2. 事件

事件是对象行为的描述，当外来动作作用于某个对象时，用户可以确定是否通过事件响应该动作。事件是一种特定的操作，在某个对象上发生或对某个对象发生。Access 2003 可以响应多种类型的事件：鼠标单击、数据更改、窗体打开或关闭及许多其他类型的事件。事件的发生通常是用户操作的结果。例如，单击某个命令按钮，该命令按钮会响应单击事件，作出相应动作。

使用事件过程或宏，可以为在窗体或控件上发生的事件添加自定义的事件响应，这里先介绍使用事件过程，宏在以后章节介绍。

在窗体设计视图中，每当使用“工具箱”向窗体添加某个控件后，可设置该控件的事件响应，设置方法有多种，常用的设置方法是：右击该控件，在弹出的快捷菜单中选择“属性”调出该控件的属性设置对话框，选择“事件”选项卡，进入该对象的事件设置界面。例如，在窗体加入文本框后，按以上操作调出文本框控件的事件设置对话框，如图 6.25 所示。

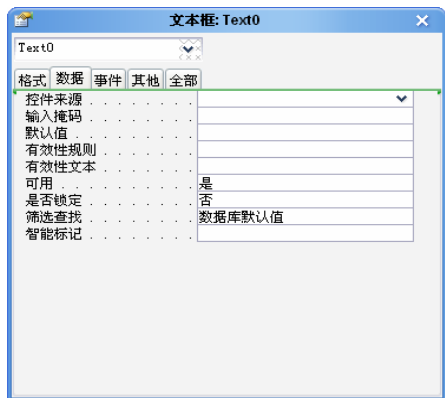


图 6.24 “文本框”控件数据属性设置对话框

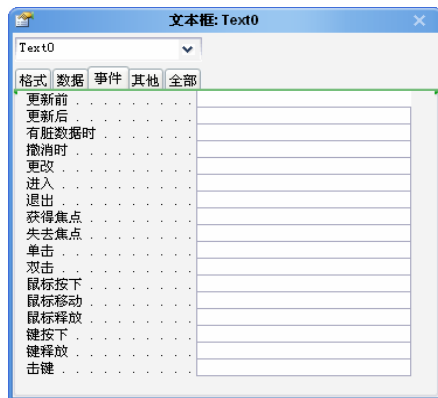


图 6.25 “文本框”控件事件设置对话框

控件事件有多种类型，各种事件的含义及添加方法窗口将在 6.3.4 小节中介绍。

3. 方法

方法是 Access 2003 提供的完成某项特定功能的操作，每种方法有一个名字，用户在系统设计中可根据需要调用方法。

例如, SetFocus 方法, 其功能是: 让控件获得焦点, 使其成为活动对象。

Access 2003 提供了多种方法, 常用方法的含义及使用方法将在 6.3.5 小节中介绍。

6.3.3 窗体与对象的属性及设置方法

1. 窗体的主要属性

窗体常用的属性:

标题(Caption): 用于指定窗体的显示标题。

默认视图(DefaultView): 设置窗体的显示形式, 可以选择单个窗体、连续窗体、数据表、数据透视表和数据透视图等方式。

允许的视图(ViewsAllowed): 指定是否允许用户通过选择“视图”菜单中的“窗体视图”或“数据表视图”命令, 或者单击“视图”按钮旁的箭头并选择“窗体视图”或“数据表视图”, 以在数据表视图和窗体视图之间进行切换。

滚动条(Scrollbars): 决定窗体显示时是否具有窗体滚动条, 属性值有 4 个选项, 分别为“两者均无”、“水平”、“垂直”和“水平和垂直”, 可以选择其一。

记录选定器(Recordselectors): 选择“是/否”, 决定窗体显示时是否有记录选定器, 即窗体最左边是否有标志块。

浏览按钮(NavigationButtons): 用于指定在窗体上是否显示浏览按钮和记录编号框。

分隔线(DividingLines): 选择“是/否”, 决定窗体显示时是否显示各节间的分隔线。

自动居中(AutoCenter): 选择“是/否”, 决定窗体显示时是否自动居于桌面的中间。

最大最小化按钮(MinMaxButtons): 决定窗体是否使用 Windows 标准的最大化 and 最小化按钮。

关闭按钮(CloseButton): 决定窗体是否使用 Windows 标准的关闭按钮。

弹出方式(PopUp): 可以指定窗体是否以弹出式窗体形式打开。

内含模块(HasModule): 指定或确定窗体或报表是否含有类模块。设置此属性为“否”能提高效率, 并且减小数据库的大小。

菜单栏(MenuBar): 用于将菜单栏指定给窗体。

工具栏(Toolbar): 用于指定窗体使用的工具栏。

节(Section): 可区分窗体或报表的节, 并可以对该节的属性进行访问。同样可以通过控件所在窗体或报表的节来区分不同的控件。

允许移动(Moveable): 在“是”或“否”两个选项中选取, 决定在窗体运行时是否允许移动窗体。

记录源(RecordSource): 可以为窗体或者报表指定数据源, 并显示来自表、查询或者 SQL 语句的数据。

排序依据(OrderBy): 为一个字符串表达式, 由字段名或字段名表达式组成, 指定排序的规则。

允许编辑(AllowEdits): 在“是”或“否”两个选项中选取, 决定在窗体运行时是否允许对数据进行编辑修改。

允许添加(AllowAdditions): 在“是”或“否”两个选项中选择, 决定在窗体运行时是否允许添加记录。

允许删除(AllowDeletions): 在“是”或“否”两个选项中选择, 决定在窗体运行时是否允许删除记录。

数据入口(DataEntry): 在“是”或“否”两个选项中选择, 如果选择“是”, 则在窗体打开时, 只显示一条空记录, 否则显示已有记录。

2. 控件属性

(1) 标签(label)控件

标题(Caption): 该属性值将成为控件中显示的文字信息。

名称(Name): 该属性值将成为控件对象引用时的标识名字, 在 VBA 代码中设置控件的属性或引用控件的值时使用。

其他常用的格式属性: 高度(Height)、宽度(Width)、背景样式(BackStyle)、背景颜色(BackColor)、显示文本字体(FontBold)、字体大小(FontSize)、字体颜色(ForeColor)、是否可见(Visible)等。

(2) 文本框(text)控件

常用的格式属性同“标签”控件。

常用的数据属性:

控件来源(ControlSource): 设置控件如何检索或保存在窗体中要显示的数据。如果控件来源中包含一个字段名, 那么在控件中显示的就是数据表中该字段的值。在窗体运行中, 对数据所进行的任何修改都将被写入字段中; 如果设置该属性值为空, 除非通过程序语句, 否则在窗体控件中显示的数据将不会被写入到数据表的字段中; 如果该属性设置为一个计算表达式, 则该控件会显示计算的结果。

输入掩码(InputMask): 用于设置控件的数据输入格式, 仅对文本型和日期型数据有效。

默认值(Default Value): 用于设定一个计算型控件或非结合型控件的初始值, 可以使用表达式生成器向导来确定默认值。

有效性规则(ValidationRule): 用于设定在控件中输入数据的合法性检查表达式, 可以使用表达式生成器向导来建立合法性检查表达式。若设置了“有效性规则”属性, 在窗体运行期间, 当在该控件中输入数据时将进行有效性规则检查。

有效性文本(ValidationText): 用于指定当控件输入的数据违背有效性规则时, 显示给用户的提示信息。

是否有效(Enabled): 用于决定能否操作该控件。如果设置该属性为“否”, 该控件将以灰色显示在“窗体”视图中, 但不能用鼠标、键盘或 TAB 键单击或选中它。

是否锁定(Locked): 用于指定在窗体运行中, 该控件的显示数据是否允许编辑等操作。默认值为 False, 表示可编辑, 当设置为 True 时, 文本控件相当于标签的作用。

(3) 组合框(combo)控件(与文本框相同的不再说明)

行来源类型(RowSourceType): 该属性值可设置为: 表/查询、值列表或字段列表, 与“行来源”属性配合使用, 用于确定可列表选择内容的来源。选择“表/查询”, “行来源”属性可设置为表或查询, 也可以是一条 Select 语句, 列表内容显示为表、查询或 Select 语

句的第一个字段内容；若选择“值列表”，“行来源”属性可设置为固定值用于列表选择；若选择“字段列表”，“行来源”属性可设置为表，列表内容将为选定表的字段名。

行来源(RowSource): 与行来源类型(RowSourceType)属性配合使用。

(4) 列表框(list)控件

列表框与组合框在属性设置及使用上基本相同，区别是列表框控件只能选择输入数据而不能直接输入数据。

(5) 命令按钮(command)控件

名字(Name): 可引用的命令按钮对象名。

标题(Caption): 命令按钮的显示文字。

标题的字体(FontName): 命令按钮的显示文字的字体。

标题的字体大小(FontSize): 命令按钮的显示文字的字号

前景颜色(ForeColor): 命令按钮的显示文字的颜色。

是否有效(Enabled): 选择“是/否”，用于决定能否操作该控件。如果设置该属性为“否”，该控件将以灰色显示在“窗体”视图中，但不能用鼠标、键盘或 TAB 键单击或选中它。

是否可见(Visible): 选择“是/否”，用于决定在窗体运行时该控件是否可见，如果设置该属性为“否”，该控件在“窗体”视图中将不可见。

图片(Picture): 用于设置命令按钮的显示标题为图片方式。

(6) 选项按钮(Option)控件、选项组(Frame)控件、复选框(Check)控件、切换按钮(Toggle)控件、选项卡控件、页控件的主要属性基本与上述控件相一致，有个别不同的将在控件设计时说明，在此不详细介绍。

3. 设置窗体属性

在 Access 2003 中，使用“属性表”和 VBE 可以查看并修改属性。用“属性表”设置属性，操作直观，但只能在设计视图状态下进行。在 VBE 中，通过命令语句可在系统运行中动态设置属性，但大部分属性可以在设计视图状态下利用“属性表”设置。

用“属性表”设置窗体属性：

(1) 在窗体设计视图中，单击窗体左上角的“窗体选定器”来选择一个窗体。

(2) 右击然后选择快捷菜单中的“属性”命令，或单击“工具栏”上的“属性”按钮，显示“属性表”。

(3) 单击需要设置其值的属性，然后执行以下操作之一：

- 在“属性”框中，键入适当的设置或表达式。
- 如果“属性”框包含箭头，单击该箭头，打开下拉列表，然后单击列表中的值。
- 如果“生成器”按钮显示在属性框的右边，单击该按钮以显示“生成器”或显示能够选择生成器的对话框。例如，可以使用“代码生成器”、“宏生成器”或“查询生成器”设置某些属性。

【例 6.5】 设置窗体的背景图案。

背景图案可以是 Windows 环境下的各种图形格式的文件，如位图文件、图元文件和图标文件等。

在窗体“格式”属性中做如下设置：

(1) “图片”属性：直接输入用作背景图案的图形文件的完整路径及文件名，或者单击生成器按钮，打开“插入图片”对话框，利用该对话框选择用作背景图案的图形文件。

(2) “图片类型”属性有 2 种选择：

- **链接**。图形文件必须与数据库同时保存，并可以单独打开图形文件进行编辑修改。
- **嵌入**。图形直接嵌入到窗体中，此方式增加数据库文件长度，嵌入后可以删除原图形文件。

(3) “图片缩放模式”属性有 3 种选择：

- **剪裁**。图形按照其实际大小直接用作窗体背景，如果图形大于窗体背景，依据窗体进行剪裁，多余部分被裁掉。此选项为默认选项。
- **拉伸**。拉伸图形以适应窗体的大小，此方式将改变图形的宽高比，除非图形的宽高比与窗体宽高比相同。
- **缩放**。改变图形大小以适应窗体背景，同时保持图形的宽高比。

(4) “图片对齐方式”属性有五种选择：

- 中心(在窗体中上下左右居中)
- 左上(窗体的左上角)
- 左下(窗体的左下角)
- 右上(窗体的右上角)
- 右下(窗体的右下角)

(5) “图片平铺”属性：是/否。根据“图片对齐方式”的设置将图形布满窗体。

【例 6.6】 设置窗体的格式属性。

默认的窗体“格式”属性设置与窗体视图如图 6.26 所示。

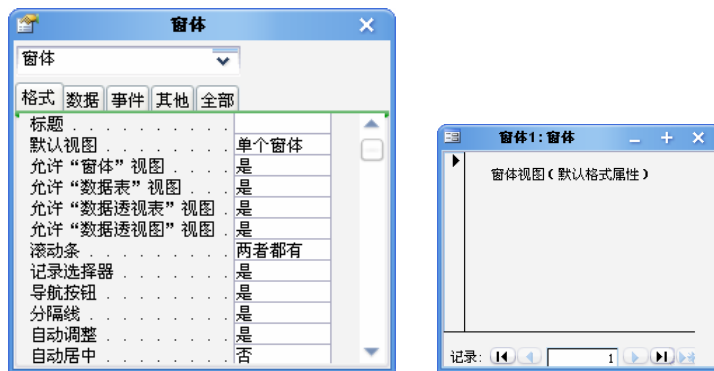


图 6.26 窗体默认“格式”属性设置与窗体视图

在本例中，设置窗体以下属性：

标题(Caption): “学生信息”

滚动条(): “两者均无”

记录选定器(): “否”

导航按钮(): “否”

分隔线(): “否”

修改以上属性后，窗体“格式”属性设置与窗体视图如图 6.27 所示。

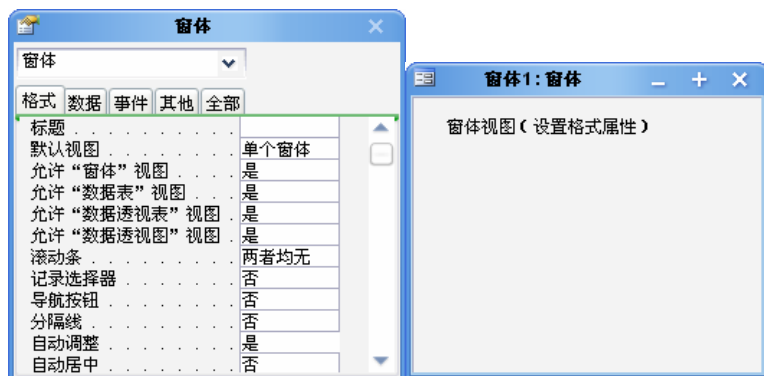


图 6.27 设置窗体“格式”属性与窗体视图

4. 在窗体中添加当前日期和时间

Access 2003 提供了将系统时钟日期和时间显示在窗体上的方法, 添加日期和时间后, 在窗体视图中, 系统时钟日期和时间将显示在窗体上。

【例 6.7】使用菜单命令为窗体添加当前日期和时间。

操作方法如下:

- (1) 进入窗体“设计”视图。
- (2) 选择“插入”菜单中的“日期和时间”命令, 打开“日期和时间”对话框。
- (3) 在“日期和时间”对话框中, 选中“包含日期”复选框, 然后选择日期格式; 选中“包含时间”复选框, 然后选择时间格式。
- (4) 单击工具栏上的“保存”按钮。

完成上述操作后, Access 2003 将在窗体上添加 2 个文本框, 其“控件来源”属性分别设置为表达式“= Date()”和“= Time()”, 如果有窗体页眉, 文本框将添加到窗体页眉节, 否则添加到主体节中。

通过以上 3 个例子, 除数据属性外, 对于窗体的其他属性设置, 可用相同的方法进行设置并查看结果。

5. 使用属性表设置控件属性

设置控件属性, 方法同窗体属性设置, 具体属性值要根据控件的具体用途来确定。

【例 6.8】建立“学生信息处理”窗体, 给出各控件的具体使用及属性设置, 结果如图 6.28 所示。

主要操作步骤如下:

- (1) 设置窗体“数据源”属性为“学生”数据表。
- (2) 完成字段对应控件的添加。

字段与控件的对应, 亦即用户要根据表字段的类型, 选择何种控件来显示、编辑字段数据。由于篇幅所限, 仅就个别情况给以说明。

图 6.28 “学生信息处理”窗体视图

添加一个“标签”控件：“标题”属性设置为“学生基本信息处理”，字体、字号作相应设置。

添加一个“文本框”控件，用于显示“学号”字段数据，系统将自动在其前添加一个“标签”控件。设置“标签”控件的“标题”属性为“学号”；设置“文本框”控件的“名称”属性为“Txh”、“控件来源”选择“学号”字段名。

添加一个“选项组”控件，在“选项组”框中加入 2 个“选项按钮”控件，用于显示学生婚否。需要说明的是：由于“学生”数据表中“婚否”字段用布尔型，所以选用“选项按钮”控件，用户亦可尝试用单“选项按钮”控件来显示“婚否”字段，查看结果有何不同。设置“选项组”控件的“控件来源”属性为“婚否”字段名；2 个“选项按钮”控件的“标题”属性分别设为“已婚”和“未婚”，“选项值”属性分别设为“-1”和“0”。在这里应注意：布尔型数据只有两个值，分别为 True 和 False，布尔型数据转换为其他类型数据时，True 转换为-1，False 转换为 0。

添加一个“组合框”控件，显示学生所在系，由于系别名称为标准内容，选用“组合框”控件，用户在录入信息时，可以选择录入。设置控件“名称”属性为“Cxb”；“控件来源”选择“系别”字段名；设置“行来源类型”属性为“表或查询”、“行来源”属性为“Select 代码名称 From dmb where 类别=“2””。这里需要说明的是：系别名称放在名为“Dmb”的代码表中，“类别”字段值为 2 时，“代码名称”字段存放内容为系别名称，代码表(Dmb)数据表视图如图 6.29 所示。

类别	代码编号	代码名称
1	1	党员
1	2	预备党员
1	3	团员
2	1	计算机系
2	2	经济管理学院
2	3	生物工程系
2	4	物理系
2	5	中文系
2	6	体育系

图 6.29 代码表数据表视图

使用向导方式添加命令按钮控件，可依此建立“首记录”、“上一条”、“下一条”、

“末记录”4个记录导航按钮；“删除记录”、“打印记录”和“添加记录”3个记录操作按钮；“关闭窗口”操作按钮。

关于对应“学生”数据表其他字段的控件的建立，用户可参照以上过程自己完成。

6. 在VBE中设置窗体和控件属性

窗体(Form)和控件(Control)对象都是 VBE 对象，可以在 VBE 子过程(Sub)、函数过程(Function)或事件过程中设置这些对象的属性。在此仅给出示例，详细的语法说明在后续章节中介绍。

(1) 设置窗体属性

在 VBE 代码中，可以通过引用 Forms 集合中单个窗体，其后跟随着属性的名称来设置窗体属性。

例如，若要将上例建立的“学生信息处理”窗体的 Visible 属性设置为 True (-1)，可在 VBE 代码中使用以下代码行：

```
Forms! 学生信息处理.Visible = True  
或 Forms! 学生信息处理.Visible = -1(可以是除 0 以外的任何值)
```

 **注意：** 若在“学生信息处理”窗体某个控件的事件代码中引用上述语句，可使用对象的 Me 属性，使用 Me 属性的代码比使用完整对象名称的代码执行得更快。上面语句可改写为：

```
Me.Visible = True  
或 Me.Visible = -1
```

(2) 设置控件属性

引用 Forms 对象的 Controls 集合中的控件。既可以隐式引用也可以显式引用 Controls 集合。

例如，若要将上例建立的“学生信息处理”窗体的“删除记录”命令按钮设为“变灰”，不可选用，其 Enabled 属性需设置为 False (-1)，“删除记录”命令按钮的名字为 Comdel，可在 VBE 代码中使用以下代码行：

```
Me!Comdel.Enabled = false(当前窗体事件代码中用)  
或 Forms!学生信息处理!Comdel.Enabled = false
```

6.3.4 窗体与对象的事件

在 Access 2003 中，对象能响应多种类型的事件，每种类型的事件又由若干种具体事件组成，通过编写相应的事件代码，用户可定制响应事件的操作。以下将分类给出 Access 2003 窗体、报表及控件的一些事件。

1. 窗口(Windows)事件

窗口事件是指操作窗口时引发的事件，如表 6.3 所示，正确理解此类事件发生的先后顺序，对控制窗体和报表的行为非常重要。

表 6.3 窗口 (Windows) 事件

事件属性	事件对象	事件发生情况
OnOpen	窗体和报表	窗体被打开, 但第一条记录还未显示出来时发生该事件。或虽然报表被打开, 但在打印报表之前发生
OnLoad	窗体	窗体被打开, 且显示了记录时发生该事件。发生在 Open 事件之后
OnResize	窗体	窗体的大小变化时发生。此事件也发生在窗体第一次显示时
OnUnload	窗体	窗体对象从内存撤销之前发生。发生在 Close 事件之前
OnClose	窗体和报表	窗体对象被关闭但还未清屏时发生

2. 数据 (Data) 事件

数据(Data)事件指与操作数据有关的事件, 又称操作事件, 如表 6.4 所示。当窗体或控件的数据被输入、修改或删除时将发生数据(Data)事件。

表 6.4 数据 (Data) 事件

事件属性	事件对象	事件发生情况
AfterDelConfirm	窗体	确认删除记录且记录实际上已经删除或取消删除之后发生的事件
AfterInsert	窗体	插入新记录保存到数据库时发生的事件
AfterUpdate	窗体和控件	更新控件或记录数据之后发生的事件; 此事件在控件或记录失去焦点时, 或单击菜单中的“保存记录”时发生
BeforeDelConfirm	窗体	在删除记录后, 但在 Access 2003 显示对话框提示确认或取消之前发生的事件。此事件在 Delete 事件之后发生
BeforeInsert	窗体	在新记录中键入第一个字符, 但还未将记录添加到数据库之前发生的事件
BeforeUpdate	窗体和报表	更新控件或记录数据之前发生的事件; 此事件在控件或记录失去焦点时, 或单击菜单中的“保存记录”时发生
Change	控件	当文本框或组合框的部分内容更改时发生的事件
Current	窗体	当焦点移动到一条记录, 使它成为当前记录, 或当重新查询窗体数据源时发生的事件
Delete	窗体	删除记录, 但在确认删除和实际执行删除之前发生该事件
NoInList	控件	当输入一个不在组合框列表中的值时发生的事件

3. 焦点 (Focus) 事件

“焦点”即鼠标或键盘操作的当前状态, 当窗体、控件失去或获得焦点时, 或窗体、报表成为激活或失去激活状态时, 将发生焦点(Focus)事件, 如表 6.5 所示。

表 6.5 焦点 (Focus) 事件

事件属性	事件对象	事件发生情况
OnActivate	窗体和报表	在窗体或报表成为激活状态时发生的事件

续表

事件属性	事件对象	事件发生情况
OnDeactivate	窗体和报表	在窗体或报表由活动状态转为非活动状态之前发生
OnEnter	控件	在控件实际接收焦点之前发生，此事件发生在 GotFocus 事件之前
OnExit	控件	当焦点从一个控件移动到同一窗体的另一个控件之前发生的事件，此事件发生在 LostFocus 事件之前
OnGot Focus	窗体和控件	当窗体或控件对象获得焦点时发生的事件。当“获得焦点”事件或“失去焦点”事件发生后，窗体只能在窗体上所有可见控件都失效，或窗体上没有控件时，才能重新获得焦点
OnLost Focus	窗体和控件	当窗体或控件对象失去焦点时发生的事件

4. 键盘 (Ksyboard) 事件

键盘(Keyboard)事件是操作键盘引发的事件，如表 6.6 所示。

表 6.6 键盘 (Ksyboard) 事件

事件属性	事件对象	事件发生情况
OnKeyDown	窗体和控件	在控件或窗体具有焦点时，键盘有键按下时发生该事件
OnKeyUp	窗体和控件	在控件或窗体具有焦点时，释放一个按下的键时发生该事件
OnKeyPress	窗体和控件	在控件或窗体具有焦点时，当按下并释放一个键或组合键时发生该事件

5. 鼠标 (Mouse) 事件

鼠标(Mouse)事件是用户操作鼠标引发的事件，如表 6.7 所示。鼠标事件应用较多，特别是“单击”事件，命令按钮的功能处理大多用鼠标(Mouse)事件来完成。

表 6.7 鼠标 (Mouse) 事件

事件属性	事件对象	事件发生情况
OnClick	窗体和控件	当鼠标在控件上单击时发生的事件
OnDbClick	窗体和控件	当鼠标在控件上双击时发生的事件，对窗体，双击窗体空白区域或窗体上的记录选定器时发生
OnMouseDown	窗体和控件	当鼠标在窗体或控件上，按下左键时发生的事件
OnMouseMove	窗体和控件	当鼠标在窗体、窗体选择内容或控件上移动时发生的事件
OnMouseUp	窗体和控件	当鼠标位于窗体或控件时，释放一个按下的鼠标键时发生的事件

6. 打印 (Print) 事件

在打印报表或设置打印格式时发生打印(Print)事件，如表 6.8 所示。

表 6.8 打印(Print)事件

事件属性	事件对象	事件发生情况
OnNoData	报表	设置没有数据的报表打印格式后, 在打印报表之前发生该事件。用该事件可取消空白报表的打印
OnPage	报表	在设置页面的打印格式后, 在打印页面之前发生该事件
OnPrint	报表	该页在打印或打印预览之前发生

7. Timer和Error事件

Timer 事件: 在 VB 中提供的 Timer 时间控件可以实现计时功能, 但在 VBE 中并没有直接提供 Timer 时间控件, 而是通过窗体的“计时器间隔(TimerInterval)”属性和“计时器触发(OnTimer)”事件来完成“计时”功能, “计时器间隔(TimerInterval)”属性值以“毫秒”为单位。

处理过程为: “计时器触发(OnTimer)”事件每隔 TimerInterval 时间间隔就被激发一次, 运行 OnTimer 事件过程, 这样重复不断, 可实现“计时”功能。

Error 事件: Error 事件在窗体或报表拥有焦点, 同时在 Access 中产生了一个运行时错误时发生。这包括 Microsoft Jet 数据库引擎错误, 但不包括 Visual Basic 中的运行时错误或来自 ADO 的错误。如果要在该事件发生时执行一个宏或事件过程, 请将 OnError 属性设置为宏的名称或事件过程。在 Error 事件发生时, 通过执行事件过程或宏, 可以截取 Access 错误消息而显示自定义消息, 这样可以根据应用程序传递更为具体的信息。

6.3.5 常用方法

1. AddMenu方法

功能: 执行 AddMenu 操作, 用于自定义(快捷)菜单栏或全局(快捷)菜单栏。

语法: DoCmd.AddMenu menuname, menumacroname, statusbartext

参数:

Menuname: 字符串表达式, 代表要添加到自定义菜单栏或全局菜单栏中的下拉式菜单名称。若要创建快捷访问键以使用键盘选择菜单, 在作为访问键的字母之前键入“And”符(&), 在菜单栏上的菜单名中, 该字母将带有下列线。

Menuacroname: 字符串表达式, 代表宏组名字。该宏组中包含菜单命令的宏。该参数是必选参数。

Statusbartext: 字符串表达式, 表示选择菜单时显示在状态栏中的文本。

说明: 用于自定义菜单栏或全局菜单栏里的 AddMenu 方法, 必须包含 menuname 和 menumacroname 参数。menuname 参数不是必选参数, 对于自定义快捷菜单和全局快捷菜单, 忽略该参数。statusbartext 参数是可选参数, 对于自定义快捷菜单和全局快捷菜单, 忽略该参数。

2. Beep 方法

功能：使计算机的扬声器发出“嘟嘟”声。

语法：DoCmd.Beep

说明：该方法没有参数。

3. CancelEvent方法

功能：取消事件。

语法：DoCmd.CancelEvent。

说明：该方法没有参数，CancelEvent 方法仅在作为事件的结果运行时才有效。

4. Close 方法

功能：关闭打开的对象。

语法：DoCmd.Close [objecttype, objectname], [save]

参数：

Objecttype: acDataAccess 2003Page、acDefault(默认值)、acDiagram、acForm、acMacro、acModule、acQuery、acReport、acServerView、acStoredProcedure、acTable。

Objectname: 字符串表达式，代表有效的对象名称，该对象的类型由 objecttype 参数指定。

Save: acSaveNo、acSavePrompt (默认值)、acSaveYes。如果该参数空缺，将假设为默认常量 (acSavePrompt)。

说明：如果将 objecttype 和 objectname 参数保留为空白(默认常量 acDefault 用作 objecttype 值)，则 Access 2003 将关闭活动窗口。如果指定 save 参数并将 objecttype 和 objectname 参数留为空白，则必须包含 objecttype 和 objectname 参数的逗号。

5. CodeDb 方法

功能：在代码模块中使用 CodeDb 方法可以确定 Database 对象的名称，此对象引用当前正在执行代码的数据库。

例如，可以在程序数据库的一个模块中使用 CodeDb 方法来创建引用程序数据库的 Database 对象，然后就可以打开基于程序数据库中表的记录集。

语法：Set database = CodeDb

参数：database，Database 对象变量。

说明：CodeDb 方法返回一个 Database 对象，该对象的 Name 属性为从其中调用该方法的数据库的完整路径和名称。

6. OpenForm 方法

功能：打开窗体

语法：DoCmd.OpenForm formname[, view][, filtername][, wherecondition][, datamode][, windowmode][, openargs]

参数:

Formname: 字符串表达式, 代表当前数据库中的窗体的有效名称。

View: acDesign、acFormDS、acNormal(默认值)、acPreview, acNormal 代表在“窗体”视图中打开窗体。

Filtername: 字符串表达式, 代表当前数据库中查询的有效名称。

Wherecondition: 字符串表达式, 不包含 WHERE 关键字的有效 SQL WHERE 子句。

Datamode: acFormAdd、acFormEdit、acFormPropertySettings、acFormReadOnly。

Openargs: 字符串表达式。用来设置窗体的 OpenArgs 属性。该设置可以在窗体模块的代码中使用。例如 Open 事件过程。在宏和表达式中可以引用 OpenArgs 属性。该参数仅在 Visual Basic 中使用。

说明: 语法中的可选参数可以空缺, 但必须包含参数的逗号。如果有一个或多个位于末端的参数空缺, 则在指定的最后一个参数后面不需使用逗号。

7. OpenModule 方法

功能: 打开 Visual Basic 模块

语法: **DoCmd.OpenModule** [modulename][, procedurename]

参数:

Modulename: 字符串表达式, 代表要打开的 Visual Basic 模块的有效名称。如果不设置该参数, Access 2003 将在数据库的标准模块中搜索全部由 procedurename 参数指定的过程, 并且打开包含这些过程的模块。

Procedurename: 字符串表达式, 代表用于打开模块的过程的有效名称。如果不设置该参数, 将打开模块的声明节。

说明: OpenModule 操作的两个参数必须至少设置一个。如果同时设置两个参数, 则 Access 2003 将在指定过程中打开指定的模块。如果 procedurename 参数空缺, 在 modulename 参数后面不需使用逗号。

8. OpenQuery 方法

功能: 打开数据库中的查询。

语法: **DoCmd.OpenQuery** queryname[, view][, datamode]

参数:

Queryname: 字符串表达式, 代表当前数据库中的查询的有效名称。

View: acViewDesign、acViewNormal(默认值)、acViewPreview。

Datamode: acAdd、acEdit(默认值)、acReadOnly。

说明: 此方法仅在 Access 2003 环境 (.mdb) 中才可用。如果指定 datamode 参数, 但空缺 view 参数, 那么必须包含 view 参数的逗号。如果空缺位于末端的参数, 则在指定的最后一个参数后面不需使用逗号。

9. OpenReport 方法

功能: 打开当前数据库中的报表

语法: **DoCmd.OpenReport** reportname[, view][, filtername][, wherecondition]

参数:

Reportname: 字符串表达式, 代表当前数据库中的报表的有效名称。

View: acViewDesign、acViewNormal(默认值)、acViewPreview。

Filtername: 字符串表达式, 代表当前数据库中查询的有效名称。

Wherecondition: 字符串表达式, 不包含 **WHERE** 关键字的有效 **SQL WHERE** 子句。

说明: 语法中的可选参数允许空缺, 但是必须包含参数的逗号。如果有一个或多个位于末端的参数空缺, 在指定的最后一个参数后面不需使用逗号。

10. OpenTable 方法

功能: 打开当前数据库中的表。

语法: **DoCmd.OpenTable** tablename[, view][, datamode]

参数:

Tablename: 字符串表达式, 代表当前数据库中的表的有效名称。

View: acViewDesign、acViewNormal(默认值)、acViewPreview。AcViewNormal 表示将在“数据表”视图中打开表。

Datamode: acAdd、acEdit(默认值)、acReadOnly。

说明: 如果指定了 **datamode** 参数而空缺了 **view** 参数, **view** 参数的逗号不能省略。如果位于末端的参数空缺, 在指定的最后一个参数后面不需使用逗号。

11. OpenView 方法

功能: 打开当前数据库中的视图。

语法: **DoCmd.OpenView** viewname [, viewmode] [, datamode]

参数:

Viewname: 字符串表达式, 代表当前数据库中视图的名称。

Viewmode: acView、Normal(默认值)、acViewDesign、acPreview。

Datamode: acEdit(默认值)、acAdd、acReadOnly。

12. Quit 方法(Application 对象)

功能: 退出 Microsoft Access 2003。在退出前, 可以从几个选项中选择一项来保存数据库对象。

语法: **Application.Quit** [option]

参数:

Option: 固有常量, 指定退出 Access 2003 时怎样处理未保存的对象。此常量可以为下列常量中的任何一个。

acSaveYes(默认值)表示保存所有对象, 不显示对话框。

AcPrompt 表示显示对话框, 询问是否保存已更改但还未存盘的任何数据库对象。

acExit 表示退出 Access 2003, 不保存任何对象。

说明: 使 **Quit** 方法和单击“文件”菜单中的“退出”命令效果相同。可以创建自定义

菜单命令或在窗体上创建一个命令按钮，此命令按钮的过程中包括了 **Quit** 方法。例如，可以将一个 **Quit** 按钮放置在窗体上，并在按钮的 **Click** 事件中包含一个使用 **Quit** 方法的过程，此方法的 **option** 参数设置为 **acSaveYes**。

13. Quit 方法

功能：DoCmd 对象的 **Quit** 方法执行 Visual Basic 中的 **Quit** 操作。

语法：**DoCmd.Quit** [options]

参数：

Options: **acQuitPrompt** **acQuitSaveAll**(默认值)**acQuitSaveNone**。

说明：增加 DoCmd 对象的 **Quit** 方法是为了提供在 Microsoft Access 2003 for Windows 95 的 Visual Basic 代码中执行 **Quit** 操作的兼容性。建议使用 Application 对象的 **Quit** 方法来代替它。

14. Refresh 方法

功能：刷新窗体对象，**Refresh** 方法用于立即刷新指定窗体或数据表中基础数据来源中的记录，以反映您或多用户环境下的其他用户对数据的更改。

语法：**Form.Refresh**

参数：**Form**，Form 对象，代表要刷新的窗体。

说明：使用 **Refresh** 方法和单击“记录”菜单中的“刷新”命令等效。**Refresh** 方法只显示对当前集中的记录所作的更改。

15. Run 方法

功能：使用 **Run** 方法可以执行一个特定的 Access 2003 或用户定义的 **Function** 或 **Sub**。例如，可以从 ActiveX 组件中使用 **Run** 方法来执行一个在某个 Access 2003 数据库中定义过的子程序。

语法：**application.Run** procedure [, arg1, arg2, ..., arg30]

参数：

Application: Application 对象。

Procedure: 要运行的 **Function** 或 **Sub** 过程的名称。

Arg1, arg2, ...: 可选。指定的 **Function** 或 **Sub** 过程的参数。最多可以有 30 个参数。

16. RunCommand 方法

功能：使用 **RunCommand** 方法执行内置菜单或工具栏命令。

语法：**[object.]RunCommand** command

参数：

Object: 可选参数，Application 对象或 DoCmd 对象。

Command: 固有常量。指定要执行的内置菜单或工具栏命令。

说明：Access 2003 中的每个菜单和工具栏命令都有一个相关的常量，在 Visual Basic 中，可以用 **RunCommand** 方法执行该常量对应的那条命令。

17. RunMacro 方法

功能: 运行 Visual Basic 中的宏操作。

语法: **DoCmd.RunMacro** macroname[, repeatcount][, repeatexpression]

参数:

Macroname: 字符串表达式, 代表当前数据库中的宏的有效名称。

Repeatcount: 数值表达式, 是一个整型值, 代表宏将运行的次数。

Repeatexpression: 数值表达式, 在每一次运行宏时进行计算。当结果为 **False (0)** 时, 停止运行宏。

说明: 如果指定 **repeatexpression** 参数, 但 **repeatcount** 参数空缺, 则必须包含 **repeatcount** 参数的逗号。如果位于末端的参数空缺, 在指定的最后一个参数后面不需使用逗号。

18. RunSQL 方法

功能: 在 Visual Basic 操作查询中使用 **RunSQL** 方法执行 SQL 操作。此方法只在 Access 2003 数据库(.mdb) 中可用。

语法: **DoCmd.RunSQL** sqlstatement[, usetransaction]

参数:

Sqlstatement: 字符串表达式, 代表操作查询或数据定义查询的 SQL 语句。

Usetransaction: 该参数为 **True (-1)** 时, 将在事务处理中包含该查询。如果不想使用事务处理, 可将该参数设置为 **False (0)**。如果该参数空缺, 将假设为默认值 (**True**)。

说明: 如果 **usetransaction** 参数空缺, 在 **sqlstatement** 参数后面不要使用逗号。

19. Save 方法

功能: 保存对象

语法: **DoCmd.Save** [objecttype, objectname]

参数:

Objecttype: **acDataAccess 2003PageacDefault**(默认值)、**acDiagram**、**acForm**、**acMacro**、**acModule**、**acQuery**、**acReport**、**acServerView**、**acStoredProcedure**、**acTable**。

Objectname: 字符串表达式, 代表由 **objecttype** 参数所选择的类型的对象名称。

说明: 如果 **objecttype** 和 **objectname** 参数空缺(对于 **objecttype** 参数, 空缺时将假设为默认常量 **acDefault**), Access 2003 将保存活动的对象。如果 **objecttype** 参数空缺, 但在 **objectname** 参数中输入了名称, 则 Access 2003 使用指定的名称保存活动的对象。如果在 **objecttype** 参数中输入了对象类型, 就必须在 **objectname** 参数中输入一个已有的对象名称。如果 **objecttype** 参数空缺, 而在 **objectname** 参数中输入名称, 则必须包含 **objecttype** 参数的逗号。

20. SetFocus 方法

功能: 使用 **SetFocus** 方法将焦点移动到指定的窗体或活动窗体的指定控件上, 或者活

动数据表的指定字段上。

语法: **Object.SetFocus**

参数:

Object 为 **Form** 对象(代表窗体)或 **Control** 对象(代表激活窗体或数据表上的控件)。

说明: 要让指定字段或控件具有焦点, 以便所有的用户输入都针对这个对象时, 可以使用 **SetFocus** 方法。

要读取一个控件的一些属性, 此控件必须具有焦点。例如, 在能读取文本框的 **Text** 属性之前, 此文本框必须具有焦点。

某些属性只有在控件没有焦点时才能设置。例如, 当控件具有焦点时, 不能将此控件的 **Visible** 或 **Enabled** 属性设置为 “False(0)”, 只能将焦点移动到可见的控件或窗体上。如果控件的 **Enabled** 属性设置为 “False”, 就不能将焦点移动到这个控件上。

如果窗体包含了 **Enabled** 属性设置为 “True” 的控件, 就不能将焦点移动到窗体本身, 而只能将焦点移动到窗体上的控件上。在这种情况下, 如果使用 **SetFocus** 将焦点移动到窗体, 焦点将移动到窗体中上次接收焦点的控件上。

21. Undo 方法

功能: 当一个控件或窗体的值已经被改变时, 可以使用 **Undo** 方法进行重置。例如, 可以使用 **Undo** 方法来清除对某个包含无效输入项的记录的一个改变。

语法: **Object.Undo**

参数: **Object** 为 **Form** 对象或 **Control** 对象。

说明: 如果 **Undo** 方法应用于窗体, 那么将失去对当前记录的所有修改。如果 **Undo** 方法应用于控件, 仅影响控件本身。

这个方法必须在更新窗体或控件前应用。可以在窗体的 **BeforeUpdate** 事件或控件的 **Change** 事件中包含这个方法。

6.3.6 常用控件的创建方法

控件是窗体设计的主要对象, 其功能主要用于显示数据和执行操作。根据应用的类型, 控件可划分为结合型、非结合型与计算型。结合型控件有数据源, 主要用来显示、输入及更新数据表中的字段, 例如文本框、组合框、列表框等控件可作为结合型控件使用; 非结合型控件没有数据源, 主要用来显示提示信息、线条、矩形及图像, 例如标签、线条、矩形及图像等控件; 计算型控件以表达式作为数据源, 表达式可以使用窗体或报表所引用的表或查询中的字段数据, 也可以是窗体或报表上其他控件的值, 例如, 文本框亦可用来作计算控件使用, 像显示 “合计” 值等。

在以上几节中主要介绍了窗体及控件的属性和事件及 Access 2003 的一些常用方法。本节将继续通过 “学生基本信息录入” 窗体, 说明窗体及控件的使用及功能按钮的设计方法。

1. 命令按钮

命令按钮是用于接受用户操作指令、控制程序流程的主要控件之一, 用户可以通过它

指示 Access 2003 进行特定的操作。命令按钮响应用户的特定动作, 这个动作触发一个事件操作, 该事件操作可以是一段程序或对应一些宏, 用于完成特定的任务。

在 Access 2003 中, 可以利用向导创建命令按钮, 也可以手工创建命令按钮。

(1) 利用向导

使用向导可方便地创建数据编辑、处理等常用功能的命令按钮, 用户不必自写处理代码, 但处理功能较弱。

【例 6.9】 通过向导创建“学生基本信息录入”窗体的命令按钮。

主要操作步骤如下:

① 打开“学生基本信息录入”窗体的“设计”视图, 确保工具箱中的“控件向导”按钮已经按下。

② 单击工具箱中的“命令按钮”按钮。在窗体需要放置命令按钮的位置单击一下, 打开“命令按钮向导”对话框之一, 如图 6.30 所示。

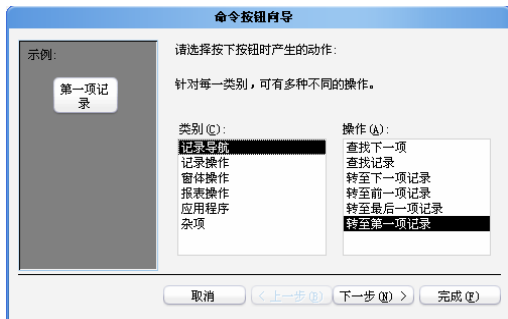


图 6.30 “命令按钮向导”对话框之一

③ 在此窗口中, Access 2003 提供了 30 多种操作, 本例中, “类别”选择“记录导航”, “操作”选择“转至第一项记录”, 单击“下一步”按钮, 打开“命令按钮向导”对话框之二, 如图 6.31 所示。

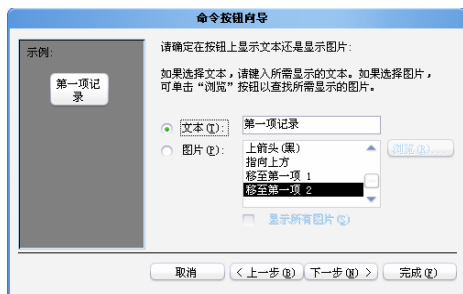


图 6.31 “命令按钮向导”对话框之二

④ 在此对话框中, 可以设置按钮上的显示内容(等同于设置按钮的“标题”属性), 可选择“文本”或“图片”。选“文本”, 在文本框中输入要在按钮上显示的内容; 选“图片”, 可单击“浏览”按钮查找所需显示的图片。单击“下一步”按钮, 打开“命令按钮向导”对话框之三, 如图 6.32 所示。

⑤ 在该对话框中, 可以为创建的命令按钮命名一个名字(等同于设置按钮的“名字”

属性), 以便以后引用。

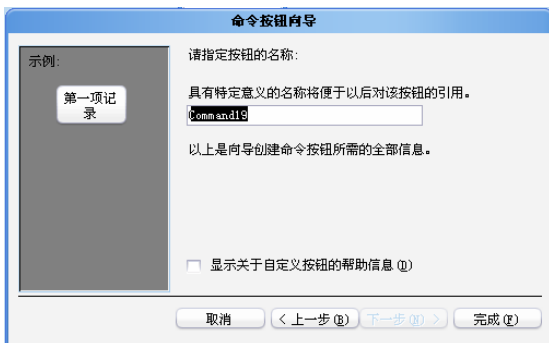


图 6.32 “命令按钮向导”对话框之三

⑥ 单击“完成”按钮, 完成该命令按钮的创建。

其他功能的命令按钮, 如“转至前一项记录”、“转至下一项记录”、“转至最后一项记录”、“添加记录”、“保存记录”、“删除记录”“打印记录”等, 创建方法与此相同。

(2) 手工创建命令按钮

手工创建命令按钮, 通过事件代码处理, 可使命令按钮具有更强的功能、更多的灵活性。其方法是: 首先将命令按钮放置在窗体中, 然后通过命令按钮的属性设置及事件代码编写, 来达到用户特定的目的。

创建步骤如下:

① 打开“学生信息处理”窗体的“设计”视图, 确保工具箱中的“控件向导”按钮已经弹起。

② 单击工具箱中的“命令按钮”按钮, 在窗体中单击要放置命令按钮的位置。

③ 设置属性: 在该命令按钮上右击, 从快捷菜单中选择“属性”, 打开属性设置对话框, 设置该命令按钮相应的属性, 例如“标题”和“名字”属性。

④ 事件过程设计: 有两种方法进入事件过程设计。

一是在该命令按钮上右击, 从快捷菜单中选择“事件生成器”, 进入如图 6.33 所示对话框, 选择“代码生成器”, 进入 VBA 代码处理窗口, 如图 6.34 所示。关于代码设计将在第 10 章介绍。

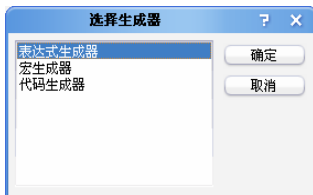


图 6.33 “事件生成器”对话框

二是在该命令按钮上右击, 从快捷菜单中选择“属性”, 打开属性设置对话框, 选择“事件”选项卡, 如图 6.35 所示, 所列项目即是命令按钮可响应的事件。

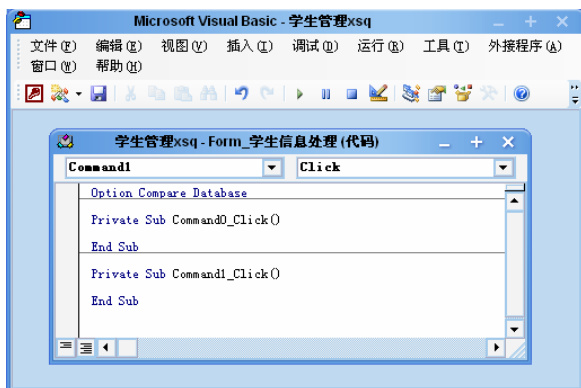


图 6.34 VBE “代码生成器”窗口



图 6.35 命令按钮事件属性设置对话框

每个事件选择项可以使用下拉箭头“~”选择“宏”或“事件过程”选项，亦可单击表达式生成器...按钮。若建有宏，可以直接选择“宏”，按钮的单击事件将执行选择的宏操作；若选择“事件过程”，然后单击表达式生成器按钮，可直接进入 VBA 代码生成器窗口；若直接单击表达式生成器按钮，则进入“选择生成器”对话框。

2. 列表框和组合框

列表框是由数据行组成的列表，每行可以包含一个或多个字段，就是说列表框可以包含多列数据，用户可以从列表框中选择某行数据。组合框是一个文本框与一个列表框的组合，在组合框中，用户既可以从列表中选择数据，也可以在文本框中输入数据。

列表框和组合框都可分为绑定的与非绑定的。绑定的列表框和组合框将选定的数据(组合框还包括输入的数据)与数据源绑定，用户选择某一行数据或输入某一数据后，该数据被保存到数据源中。

列表框和组合框有使用向导和不使用向导两种创建方法。

(1) 使用向导创建组合框

【例 6.10】以“学生管理”数据库为例，在“学生信息录入”窗体中，利用向导创建处理“系别”字段的组合框。

主要操作步骤如下：

- ① 在“学生信息录入”窗体“设计”视图下，确保“工具箱”窗口中的“控件向导

按钮”已经按下，单击“组合框”控件按钮，然后在窗体中相应的位置单击，打开“组合框向导”对话框之一，如图 6.36 所示。

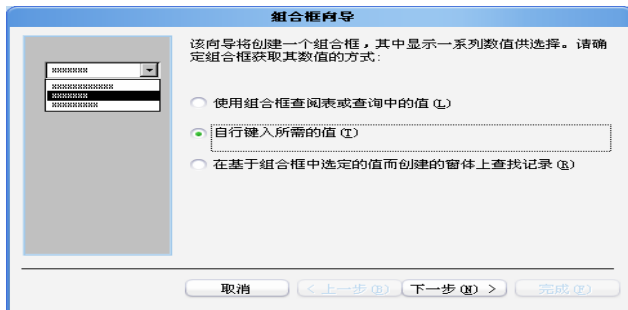


图 6.36 “组合框向导”对话框之一

② 在此对话框中有三个选项，请执行下列操作之一。

- 如果想显示固定值，则单击“自行键入所需的值”。
- 如果想显示记录源中的当前数据，则单击“使用列表框查阅表或查询中的值”或“在基于组合框中选定的值而创建的窗体上查找记录”，他们的区别在于是否对记录进行筛选。

本例中选择“自行键入所需的值”后单击“下一步”按钮，进入“组合框向导”对话框之二，如图 6.37 所示。

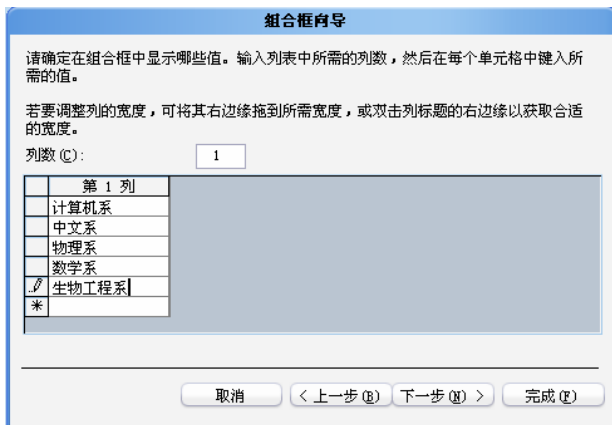


图 6.37 “组合框向导”对话框之二

③ 在图 6.37 所示对话框中，依次输入系别名称，然后单击“下一步”按钮，打开“组合框向导”对话框之三，如图 6.38 所示。

④ 在图 6.38 所示对话框中，确定组合框中选择数值后 Access 的动作，如果选择“记忆该数据供以后使用”，则创建一个非绑定的组合框，其数据由程序自由使用；如果选择“将该数据值保存在这个字段中”，则创建一个绑定的组合框，组合框的数据会自动保存到用户选择的字段中。本例中选择“将该数据值保存在这个字段中”并选择保存在字段“系别”中，完成后单击“下一步”按钮，打开“组合框向导”对话框之四，如图 6.39 所示。

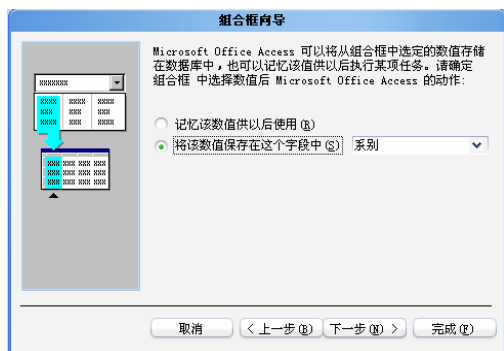


图 6.38 “组合框向导”对话框之三



图 6.39 “组合框向导”对话框之四

⑤ 用户在本对话框中指定组合框的标签显示文本，本例中键入“系列：”，单击“完成”按钮，组合框创建成功。用户还可以手工调整该组合框的属性。

(2) 不使用向导创建组合框

【例 6.11】以“学生管理”数据库为例，在“学生信息录入”窗体中，不使用向导创建处理“系列”字段的组合框。

主要操作步骤如下：

① 首先将“工具箱”窗口中的“控件向导”按钮弹起。单击“工具箱”中的“组合框”控件按钮，在窗体要放置组合框的位置单击，放入组合框。

② 右击组合框并从快捷菜单中选择“属性”命令，打开“属性”设置对话框并选择其中的“其他”选项卡，将“名称”属性改为“Cxb”，以后可以通过“名称”属性对该控件进行引用，如图 6.40 所示。

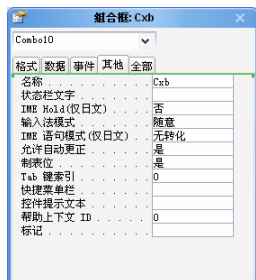


图 6.40 组合框属性设置对话框的“其他”选项卡

③ 选择“数据”选项卡，在“控件来源”中输入“系别”或选择“系别”字段，这是数据目的地，组合框选中的数据或输入的数据将保存在“系别”字段中，注意：若“控件来源”没有选择项出现，说明窗体未设置“记录源”属性；在“行来源类型”中选择“表/查询”，“行来源”中输入“Select 代码编号, 代码名称 From dmb where 类别='2'”，这两个属性决定了组合框中的列表数据的来源；在“绑定列”文本框中输入 2，确定哪个列的数据被保存起来，这里“2”代表第 2 列：“代码名称”，如图 6.41 所示。

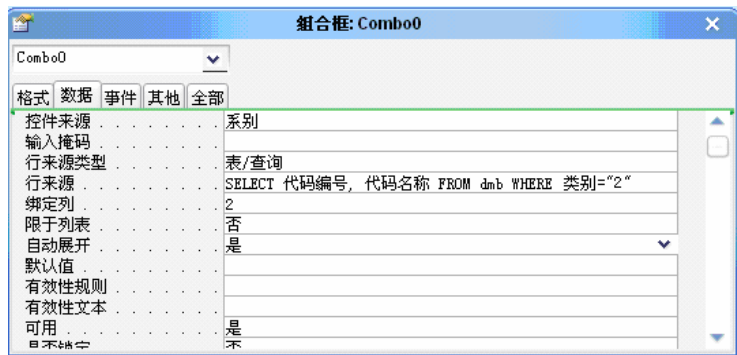


图 6.41 组合框属性设置对话框的“数据”选项卡

④ 选择“格式”选项卡，“列数”属性可以指定设置对话框的“数据”选项卡列表框或组合框的列表部分所显示的列数，在其中输入 2，表示显示两列数据，但是其中一列数据(“代码编号”列)隐藏，为此需要在属性“列宽”中输入“0 厘米;6 厘米”，这表示将第 1 列(“代码编号”)数据列隐藏，第 2 列(“代码名称”)数据列宽度是 6 厘米，如图 6.42 所示。

⑤ 最后还要将与该组合框相连的标签的文本内容改为“系别：”，完成以上操作后，组合框的属性设置完成。用户还可以根据需要调整其他的属性。

列表框的创建与组合框的创建操作相同，在此不再给出详细操作说明。



图 6.42 组合框属性设置对话框的“格式”选项卡

3. 创建选项卡控件

当窗体中的内容较多无法在一页中全部显示时，可以使用选项卡控件来进行分页显示，用户只需要单击选项卡上的标签，就可以进行页面的切换。

【例 6.12】 创建“学生信息浏览”窗体，在窗体中使用选项卡控件，一个页面显示“学生基本信息”，另一个页面显示“学生选课成绩”信息。

在窗体中使用“选项卡”控件，在选项卡中使用“列表框”控件显示学生信息。

先使用向导建立选项卡控件，主要操作步骤如下：

(1) 进入窗体“设计”视图。

(2) 单击工具箱中的“选项卡控件”按钮，在窗体中选择要放置“选项卡”的位置单击，调整其大小。系统默认“选项卡”为 2 个页，用户可根据需要使用鼠标右键插入新页。

(3) 打开“属性”对话框，分别设置页 1 和页 2 的“标题”格式属性为“学生基本信息”和“学生选课成绩”，如图 6.43 所示。

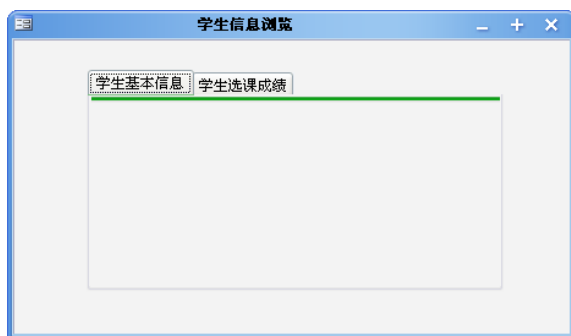


图 6.43 使用“选项卡控件”的窗体视图

(4) 确保工具箱中的“控件向导”工具按钮已按下，在“学生基本信息”页面上添加一个“列表框”控件，用来显示学生基本信息的记录内容，操作步骤如下：

① 选中“学生基本信息”页面，单击工具箱中的“列表框”控件，在“学生基本信息”选项卡的合适位置单击放入，系统显示“列表框向导”对话框之一，如图 6.44 所示。在此选择“使用列表框查阅表或查询中的值”。

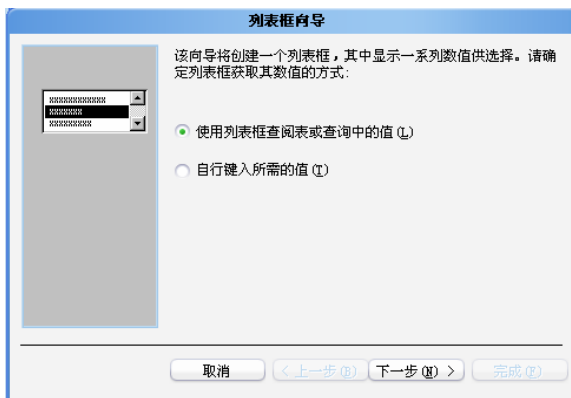


图 6.44 “列表框向导”对话框之一

② 单击“下一步”按钮，进入“列表框向导”对话框之二，如图 6.45 所示。选择为列表框提供数值的表或查询，在此选择“学生”表。

③ 单击“下一步”按钮，进入“列表框向导”对话框之三，如图 6.46 所示。选择列

表框要显示的表或查询中的字段。

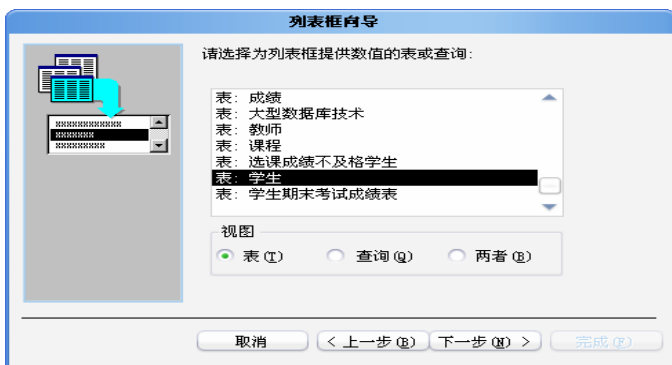


图 6.45 “列表框向导”对话框之二

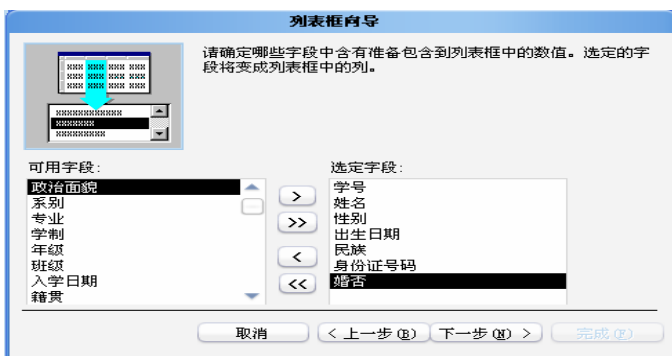


图 6.46 “列表框向导”对话框之三

④ 单击“下一步”按钮，进入“列表框向导”操作的第四个对话框，如图 6.47 所示。在列表框中显示出所有选择的表或查询中字段的列表，拖动各列右边框可以调整列的显示宽度。

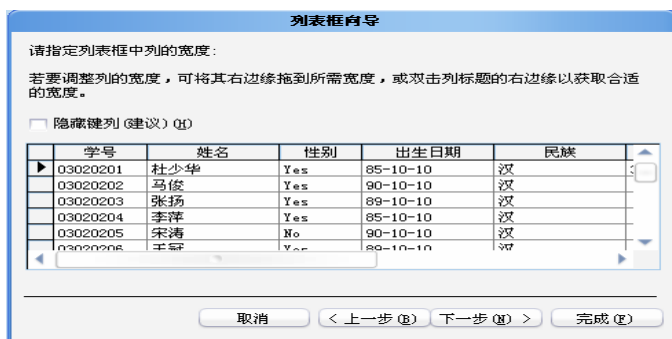


图 6.47 “列表框向导”对话框之四

⑤ 单击“完成”按钮，删除列表框的标签，适当调整列表框的大小。

⑥ 设置列表框的列标题属性，用于在列表框中显示列标题。单击列表框属性的“格式”选项卡，在“列标题”属性行中选择“是”。

用同样的方法，可在“学生选课成绩”选项卡中建立列表框，用来显示学生成绩信息。

由于建立选项卡控件主要是列表框控件的创建,使用设计视图建立选项卡控件的方法与组合框控件的建立方法基本相同,在此不做详细介绍。

4. 创建图像控件

图像控件主要用于美化窗体,可以放置开发单位的图标等。图像控件的创建比较简单,单击工具箱中的“图像”控件,在窗体的合适位置上单击,系统提示“插入图片”窗口,如图 6.48 所示,选择要插入的图片文件即可。



图 6.48 “插入图片”对话框

5. 创建选项组控件

选项组控件可以为用户提供必要的选择选项,用户只需进行简单的选取即可完成数据的录入,在操作上更直观、方便。“选项组”中可以包含复选框、切换按钮或选项按钮等控件。选项组控件的创建有使用向导和设计视图两种方法。

需要说明的是:使用选项组控件实现数据表字段的数据录入,要根据字段的类型来确定设计方法,例如“性别”字段,其类型可以是布尔型(True/False)、数据型(值为 1 和 2)和字符型(男/女)。若是布尔型或数据型,可以使用选项组控件;若是字符型,则不能使用选项组控件,可以使用组合框控件。

【例 6.13】 设“学生”表中的“性别”字段为布尔型或数据型,使用设计视图创建选项组控件,实现“性别”字段的数据录入。

主要设计步骤如下:

- (1) 进入窗体“设计”视图,设置窗体的“记录源”属性为“学生”表。
- (2) 单击工具箱中的“选项组控件”按钮,在窗体中要放置“选项组控件”的位置单击,调整其大小。
- (3) 单击工具箱中的“选项按钮”,在窗体中“选项组控件”框内单击,依次放入两个“选项按钮”,结果如图 6.49 所示。
- (4) 设置“选项组”的标签“标题”属性为“性别”,“控件来源”属性为“性别”字段。
- (5) “选项按钮”属性的设置,首先分别在格式属性中设置 2 个“选项按钮”的“标题”为“男”和“女”;然后要根据“性别”字段的类型设置“选项按钮”数据属性中“选

项值”。

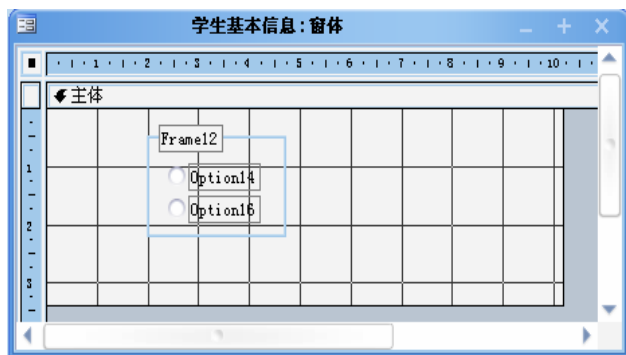


图 6.49 “选项组”控件设计窗口

若“性别”字段的类型为布尔型(True/False), 即“True”对应“男”, “False”对应“女”, 则标题为“男”的“选项按钮”的“选项值”设为-1、为“女”的“选项按钮”的“选项值”设为 0。

若“性别”字段的类型为数据型(值为 1 和 2), 即“1”代表“男”, “2”代表“女”, 则标题为“男”的“选项按钮”的“选项值”设为 1、为“女”的“选项按钮”的“选项值”设为 2, 设置窗口如图 6.50 所示。

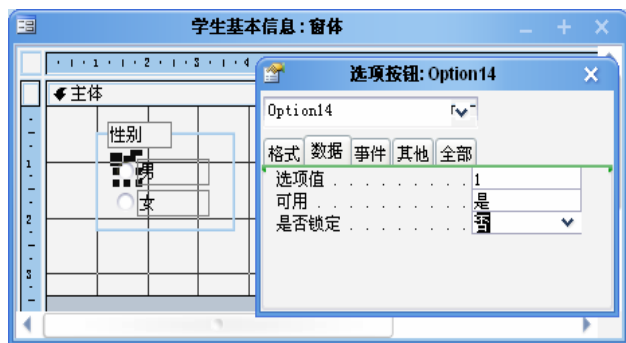


图 6.50 “选项按钮”控件“选项值”设置窗口

(6) 保存设置, 完成“选项组”控件的创建。

6. 添加ActiveX控件

Access 2003 提供了功能强大的 ActiveX 控件, 可直接在窗体中使用 ActiveX 控件添加并显示一些具有某一功能的组件, 例如日历控件等。

添加 ActiveX 控件的方法如下:

(1) 在窗体“设计”视图中, 单击工具箱中的“其他控件”按钮, 系统显示 ActiveX 控件列表。

(2) 从中选取需要的 ActiveX 控件, 例如“日历控件”。

(3) 在窗体上要放置“日历控件”的位置单击, 调整其大小, 结果如图 6.51 所示。

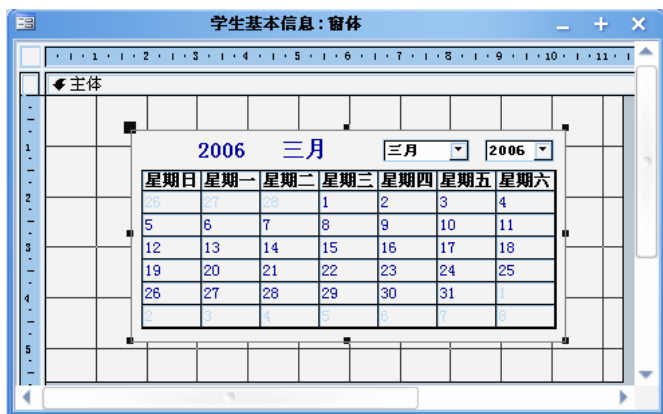


图 6.51 添加“日历控件”设计窗口

7. 删除控件

窗体中添加的每个控件都被看作独立的对象，在设计视图中，可以用鼠标选中并操作控件。例如使用选中控件后四周出现的控制句柄，可以改变控件大小、移动控件位置等。若要删除控件，可以选中要删除的控件，使用 Del 键，或选择“编辑”菜单下的“删除”命令，或使用右击快捷菜单中的“剪切”命令，该控件将被删除。

如果要删除控件附加的标签，可以只单击控件前的标签，然后删除。

6.4 窗体与控件的其他应用设计

6.4.1 创建计算控件

1. 表达式生成器

在窗体“设计”视图中，打开属性设置对话框，单击表达式生成器按钮“...”，打开“选择生成器”对话框，如图 6.52 所示。

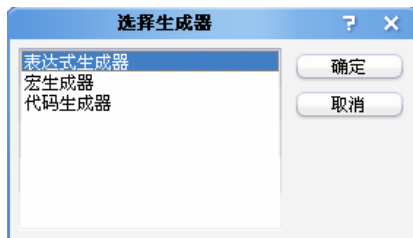


图 6.52 “选择生成器”对话框

在“选择生成器”对话框中，选中“表达式生成器”项，单击“确定”按钮，系统进入“表达式生成器”对话框，如图 6.53 所示。



图 6.53 “表达式生成器”对话框

“表达式生成器”由三部分组成，从上至下为：

(1) 表达式文本框

生成器的上方是一个表达式文本框，可在其中创建表达式。使用生成器的其他部分可以创建表达式的元素，然后将这些元素粘贴到表达式文本框中以形成表达式。也可以直接在表达式文本框中键入表达式的组部分。

(2) 运算符按钮

常用运算符的按钮位于生成器的中部。单击某个运算符按钮，“表达式生成器”将在表达式文本框中的插入点位置插入相应的运算符。单击左下角列表框中的“操作符”文件夹和中部框中相应的运算符类别，可以得到表达式中所能使用的运算符的完整列表。右侧的框列出的是所选类别中的所有运算符。

(3) 表达式元素

生成器下方含有三个列表框：

左侧的列表框列出了包含表、查询、窗体及报表等数据库对象，以及内置和用户定义的函数、常量、操作符和常用表达式的文件夹。

中间的列表框列出左侧列表框中选定文件夹内特定的元素或特定的元素类别。例如，如果在左边的列表框中单击“内置函数”，中间的列表框便列出 Microsoft Access 函数的类别。

右侧的列表框列出了在左侧和中间列表框中选定元素的值。例如，如果在左侧的列表框中单击“函数”，并在中间列表框中选定了一种函数类别，则右侧的列表框将列出选定类别中所有的函数。

 **注意：** 将标识符粘贴到表达式中时，“表达式生成器”只能粘贴在当前环境中必需的标识符部分。例如，如果从“学生信息管理”窗体的属性设置对话框中打开“表达式生成器”，然后在表达式文本框中粘贴窗体的 Visible 属性的标识符，则“表达式生成器”只粘贴属性名称：Visible。如果在窗体的环境以外使用这个表达式，则必须包含完整的标识符：Forms![学生信息管理].Visible。

2. 创建计算控件

在窗体设计中,经常需要添加一些控件,例如“文本框”控件,其显示内容不是从数据表的字段中直接取出,而是需要通过多个字段计算其值。例如工资管理窗体中的应发工资、扣发工资和实发工资等项目,这些项目一般不作为字段设计到工资数据表中,在窗体中要显示这些项目的数据,只有通过计算得出。

例如在“学生信息管理”窗体设计中,不显示学生的出生年月,要显示年龄,可以通过添加计算控件实现,操作方法如下:

(1) 进入“学生信息管理”窗体“设计”视图,添加一个“文本框”控件,命名其标签标题为“年龄”。

(2) 打开“文本框”控件的属性设置对话框,设置其“控件来源”属性。

可以使用“表达式生成器”来创建表达式,也可以自己组合表达式元素来创建表达式。

● 使用表达式生成器

在“学生信息管理”窗体中,右击“年龄”文本框,打开属性设置对话框,单击“控件来源”文本框右侧的“表达式生成器”按钮,进入“表达式生成器”对话框。

使用“函数”和“表”文件夹选项,选择生成表达式元素,结果如图 6.54 所示。



图 6.54 生成计算表达式窗口

然后按“确定”按钮,即完成“年龄”计算控件的创建。

● 使用手动方式创建

假如对函数及表达式的语法比较熟悉,可以使用手动方法创建计算表达式。在“学生信息管理”窗体中,右击“年龄”文本框,打开属性设置对话框,在其“控件来源”文本框中直接输入表达式: =Year(date())-Year([出生日期])。

完成上述设计后,在“学生信息管理”“窗体”视图中,“年龄”文本框将显示经过计算的学生年龄。需要注意的是:由于“年龄”文本框是非绑定型控件,输入或改变其值并不能保存到“出生日期”字段中。

6.4.2 查找记录

在数据表中可以查找数据。同样,在窗体中也可以使用“查找”命令来执行查找功能,具体操作方法为(以“学生信息处理”窗体为例):

(1) 在“学生管理”数据库窗口中,单击“窗体”对象。

- (2) 打开“学生信息处理”“窗体”视图。
- (3) 单击窗体中的“姓名”后的文本框，用于定位要查找的字段范围。
- (4) 选择“编辑”菜单中的“查找”命令，打开“查找和替换”对话框。
- (5) 在“查找内容”框内输入要查找学生的姓名，然后单击“查找下一个”按钮。如果找到，窗体上将显示该条记录的内容；否则，提示未找到信息。
- (6) 单击“关闭”按钮，将对话框关闭。

6.4.3 显示提示信息

“控件提示文本”(ControlTipText)属性用于设置提示文本，当鼠标指针指向控件时，将显示设置的控件提示文本。

关于该属性的说明：

- ControlTipText 属性的设置文本不能多于 255 个字符。
- 可以使用控件的属性表、宏或 Visual Basic 来设置 ControlTipText 属性。
- 对于窗体上的控件，可以使用默认控件样式或 Visual Basic 的 DefaultControl 方法来设置此属性的默认值。
- 可以在任何视图中设置 ControlTipText 属性。

ControlTipText 属性提供了一种简捷的方法来显示窗体上控件的帮助信息。

对于窗体或窗体上的控件，还有一些其他方法来提供帮助信息。StatusBarText 属性用于将控件的有关信息显示在状态栏上。如果要对窗体或控件提供更广泛的帮助，可以使用 HelpFile 和 HelpContextId 属性。

6.4.4 创建与使用主/子窗体

子窗体是窗体中的窗体，在显示具有一对多关系的表或查询中的数据时，子窗体特别有效。例如，可以创建一个带有子窗体的主窗体，用于显示“学生”表和“成绩”表中的数据。主/子窗体的数据源必须建立一对多关系，“学生”表中的数据是一对多关系中的“一”端，而“成绩”表中的数据则是此关系中的“多”端，每个同学都可以有多门选修课，如图 6.55 所示。

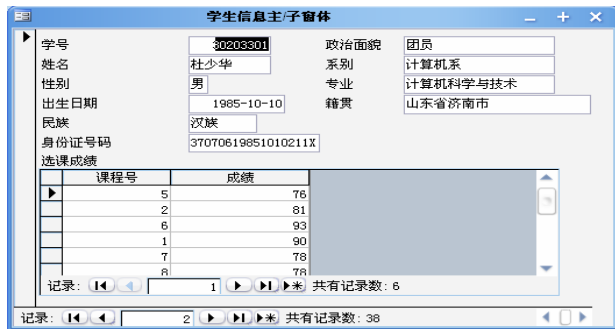


图 6.55 学生信息主/子窗体

在这类窗体中，主窗体和子窗体彼此链接，使子窗体仅显示与主窗体当前记录相关的记录。

如果用带有子窗体的主窗体来输入新记录，则在子窗体中输入数据时，Access 2003 就会保存主窗体的当前记录，这就可以保证在“多”端的表中每一记录都可与“一”端表中的记录建立联系。在子窗体中添加记录时，Access 2003 也会自动保存每一记录。

主窗体可以包含多个子窗体，还可以嵌套子窗体，最多可以嵌套七级子窗体。也就是说，可以在主窗体内包含子窗体，子窗体内可以有子窗体等。例如，可以用一个主窗体来显示学生基本信息数据，用子窗体来显示选课成绩，再用另一个子窗体来显示图书借阅信息。

要创建子窗体，可以同时创建主窗体和子窗体，也可以创建子窗体并将其添加到已有的窗体中。关于使用窗体向导创建基于多表的主/子窗体，在例 6.2 中已详细介绍。以下仅介绍将子窗体添加到已有窗体中的方法。

在添加子窗体之前，要确保已正确设置了表之间的关系。

(1) 在窗体“设计”视图中打开要向其中添加子窗体的窗体，确保已按下了工具箱中的“控件向导”按钮。

(2) 在工具箱中单击“子窗体/子报表”控件。

(3) 在窗体中要放置子窗体的位置单击。

(4) 以下操作按照向导对话框的提示进行。

(5) 最后，单击“完成”按钮后，Access 2003 将在已有的主窗体中添加一个子窗体控件，并为子窗体创建一个单独的窗体。

6.4.5 打印与预览窗体

可以在窗体的各个视图中打印窗体或预览窗体。

1. 在“设计”、“窗体”或“数据表”视图中打印窗体

(1) 在“文件”菜单中，选择“打印”命令。

(2) 在“打印”对话框中，选择需要的打印选项，单击“确定”按钮。

Access 2003 如何打印窗体，取决于要打印的视图类型，如表 6.9 所示。

表 6.9 窗体的视图类型

视图类型	窗体打印类型
“设计”视图	“窗体”视图
“窗体”视图	“窗体”视图
“数据表”视图	数据表

如果不想指定“打印”对话框中的选项而直接打印窗体，可单击工具栏上的“打印”按钮。

2. 在“数据库”窗口中打印窗体

- (1) 在“窗体”对象中，选择要打印的窗体。
- (2) 在“文件”菜单中，选择“打印”命令。
- (3) 在“打印”对话框中，选择需要的打印选项，单击“确定”按钮。

Access 2003 将在窗体的默认视图中打印窗体。

3. 在“设计”、“窗体”或“数据表”视图中预览窗体

- (1) 单击工具栏上的“打印预览”按钮。
- (2) 单击工具栏上的“视图”按钮以返回到“设计”视图、“窗体”视图或“数据表”视图中。

4. 在“数据库”窗口中预览窗体

- (1) 在“窗体”对象中，选择要预览的窗体。
- (2) 单击工具栏上的“打印预览”按钮。

Access 2003 在“打印预览”中如何显示窗体，取决于进行预览的视图类型，如表 6.10 所示。

表 6.10 打印预览时的窗体显示类型

视图类型	窗体显示类型
“设计”视图	“设计”视图
“窗体”视图	“窗体”视图
“数据表”视图	数据表
“数据库”窗口	窗体默认视图

6.4.6 设计菜单

在 Access 2003 数据库窗口中，我们会看到含有“文件”、“编辑”、“视图”、“窗口”等菜单项的主菜单栏。如果某个菜单项可以展开，包含有其他的菜单项，叫子菜单，如果它不再包含其他的菜单项叫做命令。用户可在主菜单上添加自己定义的子菜单、命令；也可以在已有的子菜单里面添加子菜单、命令；也可以定义自己的菜单栏，然后向里面添加子菜单或命令。

1. 定义自己的菜单栏

- (1) 选择“视图”菜单下的“工具栏”子菜单，然后选择“自定义”命令。显示“自定义”对话框的“工具栏”选项卡，如图 6.56 所示。
- (2) 单击“新建”按钮，打开“新建工具栏”对话框，如图 6.57 所示。
- (3) 输入工具栏名称，单击“确定”按钮，然后在“工具栏”选项卡中，选中该工具栏，单击“属性”对话框打开“工具栏属性”框，如图 6.58 所示。

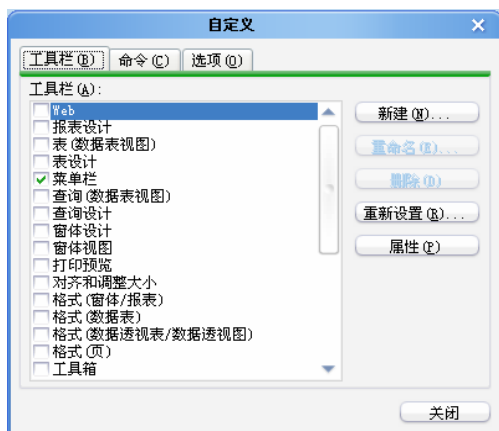


图 6.56 “自定义”对话框的“工具栏”选项卡

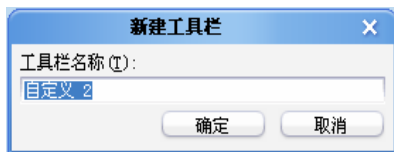


图 6.57 “新建工具栏”对话框

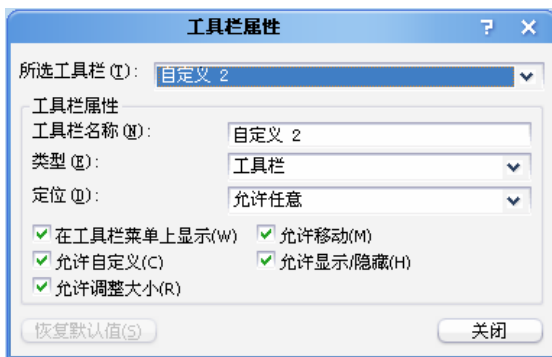


图 6.58 “工具栏属性”对话框

(4) 选择“类型”为“菜单栏”，其他选项视具体情况而定，关闭“工具栏属性”对话框，拖动新建的菜单栏停靠在标题栏下面。

2. 向菜单栏或工具栏添加自定义菜单

(1) 选择“视图”菜单下的“工具栏”子菜单，然后选择“自定义”命令，打开“自定义”对话框。

(2) 切换到“命令”选项卡，在“类别”列表框中，单击“新菜单”，系统提示如图 6.59 所示。

(3) 将“新菜单”从“命令”列表框拖动到显示的菜单栏或工具栏上，右击菜单栏或工具栏上的新菜单，在弹出的快捷菜单的“命名”文本框中键入新菜单名称，按 ENTER 键。

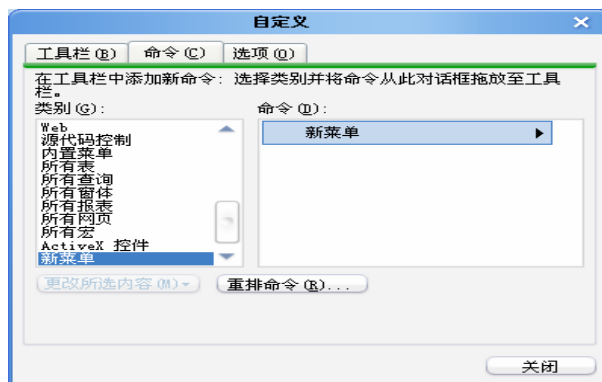


图 6.59 “自定义”对话框中的“命令”选项卡

3. 向菜单中添加命令

(1) 选择“视图”菜单中的“工具栏”菜单，然后选择“自定义”命令打开“自定义”对话框。

(2) 切换到“命令”选项卡。

(3) 在“类别”列表框中，请选择下列项目之一，如表 6.11 所示。

将需要的命令从“命令”列表框拖到菜单栏、快捷菜单工具栏或其他工具栏上。当菜单显示菜单命令的列表(新菜单是空白框)时，将鼠标指向菜单上要显示命令的地方，然后释放鼠标。

对于自定义命令要设置它的执行动作，自定义菜单也可以设置相应的动作，但一般不推荐这样做。

表 6.11 菜单命令类别

可以添加的项目	对应的单击操作
自定义命令	“文件”菜单中选“自定义”命令
内置命令	相应的菜单或视图类别
浏览 HTML 页的按钮	“Web”
操作 SourceSafe 的按钮	“源代码控制”
创建自定义控件的命令	ActiveX 控件
按照其默认视图显示窗体、报表或其他数据库对象的命令	下列之一：“所有表”、“所有查询”、“所有窗体”、“所有报表”、“所有 Web 页”
运行宏的命令	所有宏

要设置自定义命令的动作，可按以下方法操作：

- 选择“视图”菜单下的“工具栏”子菜单，然后选择“自定义”命令。
- 显示自定义命令所在的工具栏。
- 保持“自定义”对话框处于打开状态，在工具栏的按钮上右击，然后在快捷菜单中选择“属性”命令，打开如图 6.60 所示的对话框(以自定义工具栏的新菜单为例)。



图 6.60 自定义命令属性设置对话框

在“所在操作”列表框中，键入或选择要执行的动作。如果是已定义的宏，直接选择即可，如果要执行 VBA 函数，则键入所要执行的函数的名称，格式为 `=functionname()`。例如，对于名为 `SetCaption()` 的自定义函数，键入 `=SetCaption()`。对于内置函数，则键入函数的名称和其他所需的参数，如 `=MsgBox(If(Instr(Time()), "PM"), "Good Afternoon", "Good Morning"))`。

指定按钮的其他属性。

6.5 窗体外观格式设计

在窗体的“设计”视图中，有工具箱、窗体设计和格式(窗体/报表)工具栏。用户在进行设计时，需要充分利用这些工具以及窗体的弹出式菜单。

例如，可使用直线或矩形适当分隔和组织控件，对一些特殊控件使用特殊效果，对显示的文字使用颜色和各种各样的字体，均可以美化窗体。

6.5.1 加线条

利用工具箱中的“直线”和“矩形”按钮可以为窗体添加直线和矩形，然后修改其属性，将其他控件加以分隔和组织，从而大大增强窗体的可读性。

例如，要向窗体添加直线，主要操作步骤如下：

- (1) 单击工具箱中的“直线”控件按钮。
- (2) 单击窗体的任意处可以添加默认大小的直线，如果要添加任意大小的直线则可以拖动鼠标。
- (3) 单击刚添加的直线，通过拖动直线的移动手柄以调整直线的位置，选择或移动控件时按下 **Shift** 键，可保持该控件在水平或垂直方向上与其他控件对齐。可以只水平或垂直移动控件，这取决于首先移动的方向。如果需要细微地调整控件的位置，更简单的方法是按下 **Ctrl** 键和相应的方向键。以这种方式在窗体中移动控件时，即使“对齐网格”功能为打开状态，Access 2003 也不会将控件对齐网格。拖动直线的大小手柄，以调整直线的长度和角度，如果想要细微地调整窗体中控件的大小，更简单的方法便是按下 **Shift** 键，并

使用相应的方向键。要修改直线的属性，首先右击直线，从快捷菜单中选择“属性”命令，然后激活“格式”选项卡进行设置。

6.5.2 加矩形

为窗体添加矩形，其操作方法与添加直线相同，而且矩形与直线的同名属性具有相似的作用。Access 2003 为控件提供了 6 种特殊效果，即平面、凸起、凹陷、阴影、蚀刻和凿痕。其他控件如果有“特殊效果”(SpecialEffect)，属性也与此类似。

“特殊效果”属性设置影响相关的“边框样式”(BorderStyle)、“边框颜色”(BorderColor)和“边框宽度”(BorderWidth)属性设置。例如，如果特殊效果属性设为“凸起”，则忽略“边框样式”、“边框颜色”和“边框宽度”设置。另外，更改或设置“边框样式”、“边框颜色”和“边框宽度”属性会使 Access 2003 将“特殊效果”属性设置更改为“平面”。

当设置文本框的“特殊效果”属性为“阴影”时，文本在垂直方向上显示的面积会减少。可以调整文本框的“高度”(Height)属性来增加文本框的显示面积。

6.5.3 设置控件格式属性

除了如前所述的可以设置控件的特殊效果、控件上的文本颜色外，还可以通过调整控件的大小、位置等来改变窗体的布局。

1. 选择控件

选择控件包括选择一个控件和选择多个控件。要选择多个控件，首先按下 Shift 键，然后依次单击所要选择的控件。在选择多个控件时，如果已经选择了某控件后又想取消选择此控件，只要在按住 Shift 键的同时再次单击该控件即可。

通过拖动鼠标包含控件的方法选择相邻控件时，需要圈选框完全包含整个控件。如果要求圈选框部分包含时即可选择相应控件，需作进一步的设置。选择“工具”菜单中的“选项”命令，在“选项”对话框中激活“窗体/报表”选项卡，然后将“选中方式”设置成“部分包含”。通过上述设置后，当选择控件时，只要矩形接触到控件就可以选择控件，而不需要完全包含控件。

2. 移动控件

要移动控件，首先选择控件，然后移动鼠标指向控件的边框，当鼠标指针变为手掌形时，即可拖动鼠标将控件拖到目标位置。

当单击组合控件两部分中的任一部分时，Access 2003 将显示两个控件的移动控制句柄，以及所单击的控件的调整大小控制句柄。如果要分别移动控件及其标签，应将鼠标指针放在控件或标签左上角处的移动控制句柄上，当指针变成向上指的手掌图标时，拖动控件或标签可以移动控件或标签。如果指针移动到控件或其标签的边框(不是移动控制句柄)上，指针变成手掌图标时，可以同时移动两个控件。

对于复合控件，即使分别移动各个部分，组合控件的各部分仍将相关。如果要将附属

标签移动到另一个节而不想移动控件，必须使用“剪切”及“粘贴”命令。如果将标签移动到另一个节，该标签将不再与控件相关。

在选择或移动控件时按下 **Shift** 键，保持该控件在水平或垂直方向上与其他控件对齐，可以水平或垂直移动控件，这取决于首先移动的方向。如果需要细微地调整控件的位置，更简单的方法是按下 **Ctrl** 键和相应的方向键。

3. 调整控件大小

单击要调整大小的一个控件或多个控件，拖动调整大小控制句柄，直到控件变为所需的大小；也可以通过属性设置来改变控件的大小，右击所选择的控件，选择快捷菜单中的“属性”命令，在相应控件的属性设置对话框中选择“格式”选项卡，分别在“宽度”和“高度”文本框中输入控件的宽度和高度；按下 **Shift** 键，并使用相应的方向键可以细微地调整控件的大小；要调整控件的大小正好容纳其内容，则选择要调整大小的一个控件或多个控件，然后在“格式”菜单中，选择“大小”子菜单中的“正好容纳”命令，Access 2003 将根据控件内容确定其宽度和高度。

4. 对齐控件

在设计窗体时应该正确排列窗体的各控件。对齐控件包括使控件相互对齐和使用网格对齐控件两种情况。

(1) 使用网格对齐控件

首先选择要调整的控件，然后选择“格式”菜单中“对齐”子菜单中的“对齐网格”命令。

如果网格上点与点之间的距离需要调整，在设置窗体属性的对话框中选择“格式”选项卡，如果要更改水平点，为“网格线 X 坐标”属性键入一个新值。如果要更改垂直点，则为“网格线 Y 坐标”属性键入一个新值。数值越大表明点间的距离越短。网格的默认设置为水平方向每英寸 24 点，垂直方向每英寸 24 点。如果用厘米作为测量单位，则网格设置为 10×10。这些设置可以更改为 1 到 64 之间的任何整型值。如果选择了每英寸多于 24 点或每厘米多于 9 点的设置，则网格上的点将不可见。

(2) 使控件互相对齐

首先选择要调整的控件，这些控件应在同一行或同一列，然后选择“格式”菜单中的“对齐”子菜单，再选择下列其中一项命令。

- 靠左 把控件的左缘对齐最左边控件的左缘；
- 靠右 把控件的右缘对齐最右边控件的右缘；
- 靠上 把控件的上缘对齐最上面控件的上缘；
- 靠下 把控件的下缘对齐最下面控件的下缘。

如果选定的控件在对齐之后可能重叠，Access 2003 会将这些控件的边相邻排列。

5. 修改控件间隔

(1) 平均间隔控件

选择要调整的控件(至少三个)，对于有附属标签的控件，应选择控件，而不要选择其

标签。选择“格式”菜单中的“水平间距”或“垂直间距”子菜单，然后再选择“相同”命令，Access 2003 将这些控件等间隔排列。实际上只有位于中间的控件才会调整，而顶层与底层的控件位置不变。

(2) 增加或减少控件之间的间距

选择要调整的控件，选择“格式”菜单中的“水平间距”或“垂直间距”子菜单，然后再选择“增加”或“减少”命令。在控件之间的间距增加或减少时，最左侧(水平间距)及最顶端(垂直间距)的控件位置不变。

6.5.4 使用TAB键设置控件次序

在设计窗体时，特别是数据录入窗体，需要窗体中的控件按一定的次序响应键盘，便于用户操作，在“设计”视图中，Tab 键次序通常是控件的创建次序。可以使用“视图”菜单中的“TAB 键次序”命令重新设置窗体控件次序。

在“设计”视图中，打开窗体或数据访问页，执行下列操作之一：

(1) 更改窗体中的 TAB 键次序

打开窗体的“设计”视图，选择“视图”菜单中的“Tab 键次序”命令，打开“TAB 键次序”对话框，如图 6.61 所示。

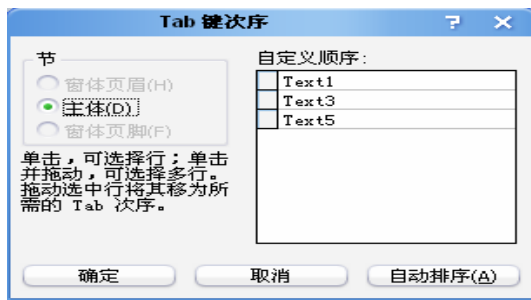


图 6.61 “Tab 键次序”对话框

如果希望 Access 2003 创建从左到右，从上到下的 Tab 键次序，单击“自动排序”。

如果希望创建自定义 Tab 键次序，在“自定义顺序”列表中，单击选定要移动的控件(单击并进行拖动可以一次选择多个控件)，然后再次单击拖动控件到列表中所需的地方。

(2) 更改数据访问页中的 TAB 键次序

打开数据访问页“设计”视图，选择要按照 Tab 键次序移动的控件，打开控件的属性设置对话框，在 TabIndex 文本框中，键入新的 Tab 键次序，如图 6.62 所示。

(3) 从 TAB 键次序中移除控件

在窗体或数据访问页“设计”视图中，选择要从 Tab 键次序中移除的控件，然后打开控件的属性设置对话框，执行下列操作之一。

- 如果控件在窗体中，则在“TabStop”(制表位)属性框中，单击“否”。只要控件的“可用”属性设为“是”，就仍可以通过单击该控件选定它。



图 6.62 设置控件的TAB键次序

- 如果控件位于数据访问页中，则将 TabIndex 属性设为 -1。只要控件的 Disabled 属性设为 False，就仍可以通过单击该控件选定它。

(4) 更改窗体中最后一个字段的 TAB 键行为

如图 6.63 所示打开窗体的属性设置对话框，在“循环”属性框中，选择下列设置：



图 6.63 窗体属性设置对话框

- 所有记录。表示在最后一个字段中按 Tab 键，焦点将移动到下一记录中的第一个字段。
- 当前记录。表示在最后一个字段中按 Tab 键，焦点将移回到当前记录中的第一个字段。
- 当前页。表示在窗体页面的最后一个字段中按 Tab 键，焦点将移回到当前页面中的第一个字段。



注意： 只能在窗体中更改最后一个字段的 Tab 键行为。

习 题

1. 什么是窗体？窗体的主要作用是什么？
2. 窗体有哪几种类型？各具有什么特点？
3. 窗体的主要创建方法有哪些？

4. 子窗体有何用处？如何建立主/子窗体？
5. 窗体的设计视图有何组成？各有什么用途？
6. 工具箱有哪些常用的控件对象？各有何用处？
7. 举例说明组合框的设计方法。
8. 举例说明列表框的设计方法。
9. 举例说明选项组的设计方法。
10. 常用的窗体格式属性有哪些？
11. 文本框控件的主要常用属性有哪些？各具有什么作用？
12. 如果要对命令按钮的“单击”事件编程，应该如何操作？
13. 在窗体数据源中可以使用多少个表或查询？为什么？