



高职高专“十一五”规划教材

SQL Server 2000

数据库应用及程序开发

➤ 孔令慧 陈 涵 主编



化学工业出版社

高职高专“十一五”规划教材

SQL Server 2000 数据库应用及程序开发

孔令慧 陈 涵 主编



化学工业出版社

· 北京 ·

本书内容主要分为三大篇：SQL Server 2000 基础篇、网络数据库应用篇和实验篇，第一篇主要介绍 SQL Server 2000 数据库管理系统的主要原理及应用；第二篇则利用开发语言 VB 和 ASP 作为前台开发语言，用 SQL Server 作为后台数据库，围绕学生管理系统和图书管理系统两个实例，全面介绍网络数据库开发的全过程；第三篇是对全书内容提供实验训练，以帮助学生消化书本知识。

本书内容丰富，层次分明，实用性和系统性强，不仅可作为高职高专信息与网络专业和计算机专业师生的教材和教学辅导书，也可作为有一定开发经验的广大编程人员的技术参考书。

图书在版编目(CIP)数据

SQL Server 2000 数据库应用及程序开发 / 孔令慧, 陈涵
主编. —北京: 化学工业出版社, 2007.8
高职高专“十一五”规划教材
ISBN 978-7-122-00868-8

I. S… II. ①孔…②陈… III. 关系数据库-数据库
管理系统, SQL Server 2000-高等学校: 技术学院-
教材 IV. TP311.138

中国版本图书馆 CIP 数据核字 (2007) 第 111772 号

责任编辑: 张建茹

文字编辑: 王听讲

责任校对: 李 林

装帧设计: 关 飞

出版发行: 化学工业出版社 (北京市东城区青年湖南街 13 号 邮政编码 100011)

印 装: 化学工业出版社印刷厂

787mm×1092mm 1/16 印张 13 1/4 字数 326 千字 2007 年 8 月北京第 1 版第 1 次印刷

购书咨询: 010-64518888 (传真: 010-64519686) 售后服务: 010-64518899

网 址: <http://www.cip.com.cn>

凡购买本书, 如有缺损质量问题, 本社销售中心负责调换。

定 价: 23.00 元

版权所有 违者必究

前 言

为了适应社会经济和科学技术迅速发展及教育教学改革的需要,全国化工高职计算机类专业教学指导委员会组织有关院校经过广泛深入的调查研究和讨论,制定了高职高专计算机类专业新一轮的教材建设规划。新的规划教材根据“以市场需求为导向,以职业能力为本位,以培养应用型高技能人才为中心”的原则,注重以先进的科学发展观调整和组织教学内容,增强认知结构与能力结构的有机结合,强调培养对象对职业岗位(群)的适应程度,对计算机类专业教材的整体优化力图有所突破,有所创新。

本课程旨在介绍现实世界中的数据在计算机中的存储方式及如何管理和组织数据。作为高职高专院校计算机专业的一门专业课程,网络数据库技术课程能够具体联系计算机基本操作、网络基础知识、数据库、数据表的设计与数据编辑、查询与报表制作、数据库中数据的一致性与安全性等具体内容。如果采用不同的平台作为开发语言,网络数据库技术课程将涉及多种计算机语言(如 Basic 语言、Asp 语言、C 语言或 C++语言),因此,本课程是一个集网络、数据库、程序设计语言为一体的综合化课程,将有助于学生学习如何借助计算机实现各种信息数据的存储、处理、加工的方法,同时也为学习其他信息类课程奠定良好的基础。

本书具有实用性强和系统性强的特点。本书的实用性体现在全书采用“围绕任务讲管理,根据需求做开发”的思想。第一篇围绕具体的项目介绍后台数据库如何建立和管理,以大量实例引导学生如何管理 SQL Server 2000;第二篇遵循“选用什么开发方案,如何实现开发环境,各模块如何设计,各模块代码如何编写”的思路,通过两个实例向学生介绍网络数据库的开发过程。第三篇本着“稳扎稳打,夯实基础”的原则,将前两部分的操作要点以实验的形式让学生多练习,以便提高操作熟练度,加深对理论知识的理解。本书的系统性体现在全书以开发两个项目为主线,系统介绍了 SQL Server 数据库的管理方法和开发技术。本书内容连贯,主线明晰,能够帮助学生全面理解 SQL Server 数据库的体系结构,并建立完整的项目开发观念。

本书中所用到的一些人名、家庭住址均为虚构,如有雷同,实属巧合。

本书内容已制作成用于多媒体教学的 PowerPoint 课件,并将免费提供给采用本书作为教材的高职高专院校使用。如有需要可到: <http://www.cip.com.cn/cbs/electronic/index.htm> 下载。

参加本书编写的人员都是在各高职高专院校从事计算机基础教学和研究的一线教学人员。本书由孔令慧、陈涵任主编,其中第一、五、六、七章由孔令慧与陈涵共同编写,第二、三、四、九章由陈涵编写,第八章由周胜编写,第十章由刘晋编写。全书由孔令慧统稿,孙吉杰主审。

本书在编写过程中得到了全国化工高职高专计算机类专业教学指导委员会和南京化工职业技术学院信息系及化学工业出版社的鼎力支持,在此谨表示衷心的感谢。本书在编写过程中参考了相关图书的一些资料,在此对这些图书的作者表示感谢。

学习本书需要先修 ASP 和 VB 课程,和本书相关联的课程还有网络基础和网络操作系统。

限于编者水平,书中不足或疏漏之处,恳请广大读者不吝指正,以便今后修正,逐步完善。

编 者
2007 年 6 月

目 录

第一篇 基础篇.....1

1 数据库基础.....1

1.1 数据库技术发展简述.....1

1.2 数据库的基本概念.....3

1.3 几种常见的数据模型.....5

1.4 关系运算.....8

1.5 关系规范化.....10

本章小结.....12

习题.....12

2 SQL Server 2000 概述.....13

2.1 SQL Server 的发展简史.....13

2.2 SQL Server 2000 的特性.....13

2.3 SQL Server 2000 的体系结构.....14

2.4 启动、暂停和关闭服务器.....15

2.5 企业管理器.....16

2.6 查询分析器.....17

2.7 联机帮助.....17

本章小结.....18

习题.....18

3 数据库、表的创建及管理.....19

3.1 数据库的创建及管理.....19

3.2 数据库的备份与还原.....26

3.3 表的创建及管理.....32

本章小结.....38

习题.....38

4 表中数据的操作.....40

4.1 数据的导入与导出.....40

4.2 表中数据的添加、删除与修改.....44

4.3 数据的完整性.....48

4.4 表中数据的查询.....55

本章小结.....67

习题.....67

5 T-SQL 语言.....68

5.1 常量与变量.....68

5.2 运算符.....71

5.3 流程控制语句.....75

5.4 常用系统函数.....79

本章小结.....88

习题.....88

6 SQL Server 2000 高级对象.....89

6.1 索引.....89

6.2 视图.....92

6.3 存储过程.....98

6.4 触发器.....108

本章小结.....114

习题.....114

7 数据库的安全性管理.....116

7.1 SQL Server 2000 身份验证方式.....116

7.2 帐号及帐号的管理.....117

7.3 用户及用户的管理.....121

7.4 权限管理.....123

7.5 角色及角色的管理.....128

本章小结.....133

习题.....133

第二篇 应用篇.....135

8 连接 SQL Server 2000.....135

8.1 在 VB 中连接 SQL Server.....135

8.2 在 ASP 中连接 SQL Server.....138

本章小结.....141

习题.....141

9 学生信息管理系统 (VB+SQL 开发).....142

9.1 概述.....142

9.2 用户登录模块的设计.....142

9.3 主窗体的设计.....145

9.4 增加用户模块设计.....147

9.5 修改密码模块设计.....149

9.6 切换用户模块设计.....150

9.7 学生信息查询模块设计.....150

9.8 学生信息录入模块设计.....154

9.9 学生信息修改模块设计	157	实验 1 熟悉 SQL Server 2000 环境及 数据库的创建	191
9.10 课程查询与设置模块设计	160	实验 2 表的创建及管理	192
9.11 选课模块设计	164	实验 3 数据完整性	193
9.12 学生成绩查询模块设计	166	实验 4 数据查询 (一)	194
本章小结	169	实验 5 数据查询 (二)	195
习题	169	实验 6 视图和索引的创建及应用	195
10 图书管理信息系统 (ASP+SQL 开发)	170	实验 7 利用 T-SQL 语句编写程序	197
10.1 系统概述	170	实验 8 存储过程的创建及应用	197
10.2 用户登录模块的设计	170	实验 9 触发器的创建及应用	198
10.3 添加图书分类模块的设计	174	实验 10 数据库备份与还原	199
10.4 图书添加模块的设计	176	实验 11 数据库的安全性管理	199
10.5 图书查询模块的设计	178	实验 12 网络数据库的连接	200
10.6 图书借阅模块的设计	184	实验 13 网络数据库的操作	201
10.7 图书归还模块的设计	187	附录	202
本章小结	190	附录 A 学生信息管理系统	202
习题	190	附录 B 图书管理信息系统	203
第三篇 实验篇	191	参考文献	205

第一篇 基础篇

信息技术是当今使用频率最高的名词之一，它在工业、农业和日常生活中得到越来越广泛的运用，而数据库技术则是信息技术中的一个重要支撑，没有数据库技术，人们在浩瀚的信息世界中将显得手足无措。从 20 世纪 50 年代中期开始，计算机应用从科学研究中渐渐扩展到社会各行各业中，因此数据库技术得到迅猛发展。现在数据库已成为信息管理、办公自动化、计算机辅助设计等应用的主要软件工具之一。本篇以 SQL Server 2000 为例，主要介绍数据库基本原理和数据库管理系统的基础知识、基本操作及日常管理，为学习本书第二篇知识打下良好基础。

1 数据库基础

随着计算机技术与网络通信技术的发展，数据库技术已成为信息社会中对大量数据进行组织与管理的重要手段，在计算机的三大主要应用领域（科学计算、数据处理和过程控制）中，数据处理是计算机应用的主要方面。数据库技术就是作为数据处理中的一门技术而发展起来的。

数据库技术研究旨在解决计算机在信息处理过程中出现的大量数据如何实现共享、减少冗余、保障安全及高效检索和处理的问题。本章主要介绍数据库的技术发展历程和数据库的相关基本概念。

1.1 数据库技术发展简述

数据库技术是计算机科学技术中发展最快的重要分支之一，已成为计算机信息系统和应用系统的重要技术支撑，自 20 世纪 60 年代末开始，计算机中数据的管理随着计算机硬件、软件技术和计算机应用范围的发展，经历了四个阶段：人工管理阶段、文件系统阶段、数据库系统阶段和高级数据库系统阶段。

1.1.1 人工管理阶段

20 世纪 50 年代以前，计算机主要用于数值计算。从当时的硬件看，外存只有纸带、卡片、磁带，没有直接存取设备；从软件看（实际上，当时还未形成软件的整体概念），没有操作系统以及管理数据的软件；从数据上看，数据量小，不保存，数据不能独立于程序，想要修改数据必须先修改程序，也必须各自定义，无法共享，存在重复数据。此阶段程序运行的结果完全需要人工管理，故有人工管理阶段之称。如图 1-1 所示。

人工管理阶段的特点为：数据不保存、没有软件管理、没有文件概念、数据面向应用。



图 1-1 数据的人工管理

1.1.2 文件系统阶段

20 世纪 50 年代后期到 60 年代中期，出现了磁鼓、磁盘等直接存取数据的存储设备。

1954 年出现了第一台用于商业数据处理的电子计算机 UNIVACI, 标志着计算机开始应用于以加工数据为主的事务处理阶段。这种基于计算机的数据处理系统也就从此迅速发展起来。

这种数据处理系统是把计算机中的数据组织成相互独立的数据文件, 系统可以按照文件的名称对其进行访问, 对文件中的记录进行存取, 并可以实现对文件的修改、插入和删除,

这就是文件系统。文件系统实现了记录内的结构化, 即给出了记录内各种数据间的关系。但是, 文件从整体来看却是无结构的, 其数据面向特定的应用程序, 虽然存在共享性, 仍很冗余, 同时数据管理和维护的代价也很大。如图 1-2 所示。

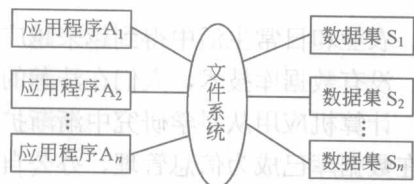


图 1-2 数据的文件系统管理

在这个阶段, 计算机已有了可直接存储的外存储器, 也有了操作系统和相关的数据管理软件。

在文件系统中, 数据按内容、结构和用途组成若干个文件, 文件一般为一个用户或用户组所有, 但也可供指定的其他用户共享。文件系统管理阶段有以下特点。

(1) 文件系统管理的优点

- ① 数据能够长期保存在磁盘上。
- ② 有了提供文件与程序之间进行存取的转换软件。
- ③ 文件形式多样。
- ④ 数据存取以记录为单位。

(2) 文件系统管理的缺点

- ① 数据冗余大。
- ② 数据易产生不一致性。
- ③ 数据和程序之间仍缺少独立性。

当数据的人工管理阶段发展到文件管理阶段时, 程序和数据已初步分开, 至此数据库系统管理已经在孕育之中。

1.1.3 数据库系统阶段

20 世纪 60 年代后期, 计算机性能得到提高, 更重要的是出现了大容量磁盘, 存储容量大大增加且价格下降。在此基础上, 为了能克服文件系统管理数据时的不足, 满足和解决实际问题中多个用户, 多个应用程序共享数据的要求, 从而使数据能为尽可能多的应用程序服务, 这就出现了数据库这样的数据管理技术。数据库的特点是数据不再只针对某一特定应用, 而是面向全组织, 具有整体的结构性, 共享性高, 冗余度小, 与数据间具有一定程序的独立性, 并且实现了对数据统一性的控制。数据库技术的应用使数据存储量猛增, 用户增加, 而且数据库技术的出现使数据处理系统的研制, 从围绕以加工数据的程序为中心转向围绕共享的数据来进行。数据库系统管理示意图如图 1-3 所示。



图 1-3 数据库系统管理

在数据库系统中,数据不再仅仅服务于某个程序或用户而是看成一个单位的共享资源,由一个叫数据库管理系统(DataBase Manage System, DBMS)的软件统一管理。

由于DBMS的统一管理,应用程序不必直接介入诸如打开、关闭、读、写文件等低级操作,而由DBMS代办,用户也不必关心数据存储在哪里,文件结构的修改也由DBMS屏蔽,用户看不到这些修改,从而减少应用程序的修改量,提高数据的独立性。由于数据库的统一管理,人们可以从全方位着眼,合理组织数据,减少数据冗余。

数据库系统阶段所具有的功能包括:数据的安全性、数据的完整性约束、并发控制、数据库的恢复。

1.1.4 高级数据库系统阶段

20世纪80年代以来关系数据库理论日趋完善,逐步取代网状和层次数据库而占领了市场,并向更高阶段发展。目前数据库技术已成为计算机领域中最重要技术之一,它是软件科学中的一个独立分支,正在朝分布式数据库、知识库系统、多媒体数据库方向发展。特别是现在的数据仓库和数据挖掘技术的发展,大大推动了数据库向智能化和大容量化的发展趋势,充分发挥了数据库的作用。

分布式数据库、面向对象和智能型数据库被称为高级数据库技术。

1.2 数据库的基本概念

数据库技术涉及许多基本概念,主要包括数据、数据处理、数据库、数据库管理系统以及数据库系统等。

1.2.1 数据

数据是指存储在某一媒体上能够识别的物理符号。所谓“符号”,不仅指数字、字母、文字和其他特殊符号,而且包括图形、图像、动画、影像、声音(包括语音、音乐)等多媒体数据,但它们都是经过数字化后存入计算机的。因此,数据的概念应包括两个方面:其一是描述事物特性的数据内容;其二是存储在某一媒体上的数据形式。

数据与信息是密切相关的,数据是信息的载体,信息是数据所表示的内容,是人们消化理解了的数据。如“95”是一个数据,它可以表示成某个年级的人数,可以表示成某人的体重,也可以理解成某学生一门课程的分数等等。可以看出将信息用某符号记录下来就成了数据,数据的形式不能完全表达其内容,它必须与特定的语义组合起来才能表示确定的信息含义。

1.2.2 数据处理

数据处理是指对各种形式的数据进行收集、存储、加工和传播的一系列活动的总称。其目的之一是从大量的、原始的数据中抽取、推导出对人们有价值的信息以作为行动和决策的依据;目的之二是为了借助计算机科学地保存和管理复杂的、大量的数据,以便人们能够方便而充分地利用这些宝贵的信息资源。早在1880年,美国在进行人口普查统计时就采用了穿孔卡片来存储普查数据,并采用机械设备来完成对这些数据的处理工作。电子计算机的出现以及其后其硬件、软件的迅速发展,加之数据库理论和技术的发展,为数据管理进入一个革命性阶段提供有力的支持。

数据处理不同于数据管理,数据管理是指数据被数字化后,对其进行分类、组织、编码、存储、检索和维护。主要是为了从原始数据中提取对人们有价值的信息,以提高工作效率。

1.2.3 数据库

数据库(DataBase, DB)是与特定的主题或目的相关的数据的集合。数据库可以直观地

理解为存放数据的仓库，只不过这个仓库是计算机的大容量存储器（例如，硬盘就是一种最常见的计算机大容量存储设备）。而且数据必须按照一定的格式存放，因为它不仅需要存放，而且还要便于查找。

可以认为数据库是被长期存放在计算机内、有组织的、可以表现为多种形式的可共享的数据集合。数据库技术使数据能按一定格式组织、描述和存储，且具有较小的冗余度（减少数据重复存储），较高的数据独立性（数据与使用它的程序相互独立）和易扩展性，并可为多个用户所共享（多个用户同时对数据进行读写）。

1.2.4 数据库管理系统

数据库管理系统（DataBase Management System, DBMS）是对数据库进行管理的系统软件，它的职能是有效地组织和存储数据、获取和管理数据，接受和完成用户提出的访问数据的各种请求。它的主要功能包括以下几个方面。

（1）数据定义功能

DBMS 提供了数据定义语言（Data Definition Language, DDL），用户通过它可以方便地对数据库中的相关内容进行定义。例如，创建数据库、表、索引等。

（2）数据操纵功能

DBMS 提供了数据操纵语言（Data Manipulation Language, DML），用户通过它可以实现对数据库的基本操作。例如，对数据库表中数据的查询、插入、删除和修改。

（3）数据库控制功能

这是 DBMS 的核心部分，它包括并发控制（即处理多个用户同时使用某些数据时可能产生的问题）、安全性检查、完整性约束条件的检查和执行、数据库的内部维护（例如，索引的自动维护）等。所有数据库的操作都要在这些控制程序的统一管理下进行，以保证数据的安全性、完整性，以及多个用户对数据库的并发使用。

（4）数据库的建立和维护功能

数据库的建立和维护功能包括数据库初始数据的输入、转换功能，数据库的转储、恢复功能，数据库的重新组织功能和性能监视、分析功能等。这些功能通常是由一些实用程序完成的。它是数据库管理系统的一个重要组成部分。

（5）数据通信功能

包括与操作系统（Operating System, OS）的联机处理、分时处理和远程作业传输等，这一功能对分布式数据库系统尤为重要。

DBMS 是一个通用的软件系统，它由一组计算机程序构成，能对数据库进行存储、安全性和完整性管理，为用户提供一个软件平台，使用户能快速建立、维护、检索、存取和处理数据库的信息。

1.2.5 数据库系统

数据库系统（Data Base System, DBS）是指拥有数据库技术支持的计算机系统，它可以实现有组织地、动态地存储大量相关数据，提供数据处理和信息资源共享服务。数据库系统不仅包括数据本身，还包括相应的硬件、软件和各类人员，如图 1-4 所示。

（1）数据库集合

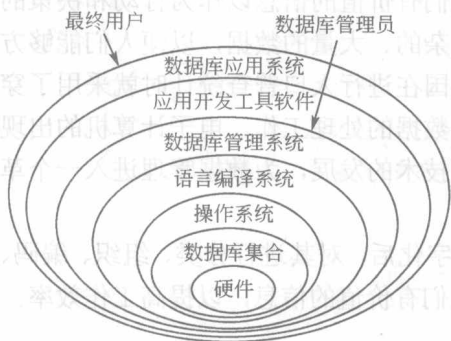


图 1-4 数据库系统组成示意图

数据库系统包括若干个设计合理、能满足应用

需要的数据库, 这些数据库由 DBMS 通过操作系统读写。

(2) 硬件系统

由于一般数据库系统数据量很大, 加之 DBMS 丰富的强有力的功能使得自身的体积很大, 因此整个数据库系统对硬件资源提出了较高的要求, 这些要求包括以下内容。

- ① 有足够大的内存以存放操作系统、DBMS 的核心模块、数据缓冲区和应用程序。
- ② 有足够大的直接存取设备存放数据 (例如、磁盘), 有足够的其他存储设备来进行数据备份。
- ③ 要求计算机有较高的数据传输能力, 以提高数据传送率。

(3) 系统软件

主要包括操作系统、数据库管理系统、与数据库接口的高级语言及其编译系统, 以及以 DBMS 为核心的应用开发工具。操作系统是计算机系统必不可少的系统软件, 也是支持 DBMS 运行必不可少的系统软件。

数据库管理系统是数据库系统不可缺少的系统软件, 它提供数据库的建立、使用和维护功能。一般来讲, 数据库管理系统的数据处理能力较弱, 所以需要与数据库接口的高级语言及其编译系统, 以便于开发应用程序。

(4) 数据库应用系统

数据库应用系统是为特定应用开发的数据库应用软件。数据库管理系统为数据的定义、存储、查询和修改提供支持, 而数据库应用系统是对数据库中的数据进行处理和加工的软件, 它面向特定应用。例如, 基于数据库的各种管理软件: 管理信息系统、决策支持系统和办公自动化等都属于数据库应用系统。

(5) 各类人员

参与分析、设计、管理、维护和使用数据库的人员均是数据库系统的组成部分。他们在数据库系统的开发、维护和应用中起着重要的作用。分析、设计、管理和使用数据库系统的人员主要是: 数据库管理员 (Data Base Administrator, DBA)、系统分析员、应用程序员和最终用户。

对于规模较大的数据库系统必须有专人全面负责建立、维护和管理数据库系统, 承担此任务的人员称为 DBA。其主要职责包括: 定义并存储数据库的内容, 监控数据库的使用, 负责数据库的日常维护以及必要时重新组织和改进数据库等。

系统分析员是建立数据库系统的主要参与人员, 负责应用系统的需求分析和规范说明。他们要和用户相结合, 确定系统的基本功能、数据库结构、应用程序的设计以及硬软件配置等, 并组织整个系统的开发。

应用程序员负责设计应用系统的程序, 为最终用户开发适用的数据库应用系统。

数据库系统的最终用户有不同层次, 一般将最终用户分为操作层、管理层、决策层。他们通过应用系统的用户接口使用数据库。常用的接口方式有菜单驱动、表格操作、图形显示、随机查询和对数据库中的数据进行统计等, 分析时使用专用的软件和分析、决策模型。

1.3 几种常见的数据模型

通俗地讲数据模型就是现实世界数据特征的模拟和抽象。在数据库中用数据模型这个工具来抽象、表示和处理现实世界中的数据和信息。

数据库中最常见的数据模型有三种, 它们是: 层次模型、网状模型、关系模型。

1.3.1 层次模型

若用图来表示,层次模型是一棵倒立的树。在数据库中,满足以下两个条件的数据模型称为层次模型:

① 有且仅有一个节点无父节点,这个节点称为根节点;

② 其他节点有且仅有一个父节点。

在层次模型中,每个节点代表一种实体类型,节点层次从根开始定义,根为第一层,根的子节点为第二层,根为其子节点的父节点,同一父节点的子节点称为兄弟节点,没有子节点的节点称为叶节点。

图 1-5 所示的抽象层次模型中,R1 为根节点;R2 和 R3 为兄弟节点,是 R1 的子节点;R4 和 R5 为兄弟节点,是 R2 的子节点;R3、R4 和 R5 为叶节点。

例如:一个学校的数据信息可以用图 1-6 的层次模型来表示。大学是根节点,每一个大学都会有各自的院系和行政办公室;每一个院系都有教研室和班级,教研室由教师组成,班级由学生构成。

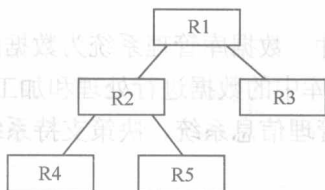


图 1-5 层次模型

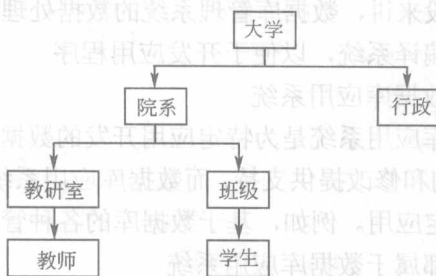


图 1-6 学校的层次模型

1.3.2 网状模型

若用图 1-5 来表示,网状模型是一个网络结构的模型,在数据库中,满足以下两个条件的数据模型称为网状模型:

① 允许一个以上的节点无父节点;

② 一个节点可以有多于一个的父节点。

图 1-7 所示的抽象网状模型中,R2 和 R3 均无父节点,R1 有两个父节点。

例如:学生和各门课程之间可以用图 1-8 的层次模型来表示。

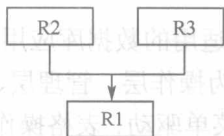


图 1-7 网状模型

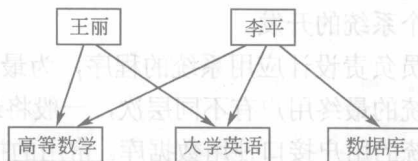


图 1-8 学生与课程的网状模型

1.3.3 关系模型

(1) 关系模型

在关系模型中,数据的逻辑结构是一张二维表。在数据库中,满足下列条件的二维表称为关系模型:

- ① 每一列中的分量是类型相同的数据;
- ② 列的顺序可以是任意的;
- ③ 行的顺序可以是任意的;
- ④ 表中的分量是不可再分割的最小数据项, 即表中不允许有子表;
- ⑤ 表中的任意两行不能完全相同。

表 1-1 所示的关系模型中, “学号”、“姓名”、“性别”和“系别”这四列中每一列的分量都是文本型的数据, “年龄”这一列的分量都是数值型的数据; 列的顺序可以是任意的; 行的顺序可以是任意的; 表中的分量是不可再分割的最小数据项; 表中的任意两行不能完全相同。

表 1-1 学生表

学 号	姓 名	性 别	年 龄	系 别	学 号	姓 名	性 别	年 龄	系 别
2005001	李明	男	19	机械系	2005003	徐菁	女	18	信息系
2005002	张小珏	女	20	化工系	2005004	张凡	男	21	信息系

(2) 关系模型的有关概念

- ① 实体 现实世界中可以相互区分, 相互识别的物体。
- ② 关系 一个关系就是一张二维表, 每个关系有一个关系名。
- ③ 元组 表中的一行即为一个元组。
- ④ 属性 (又称为字段) 表中的一列即为一个属性。
- ⑤ 主码 (又称为关键字或主键) 表中的某个属性组, 它可以唯一确定一个元组。
- ⑥ 外键 (又称为外部关键字) 当表中的某属性或属性组合不是该表的主码 (或只是主码的一部分), 但却是另一个表的主码, 称该属性或属性组合为这个表的外键。
- ⑦ 域 一个属性的取值范围称为一个域。
- ⑧ 关系模式 是表结构的描述。可以用“关系名 (字段名 1, 字段名 2, ……, 字段名 n)”来表示, 主码用下划线标出。

⑨ 关系数据库模式 一组关系模式的集合。

(3) 实体间的联系

数据库不仅描述数据 (实体) 本身, 而且描述数据 (实体) 之间的联系。在关系数据库中, 关系严格地建立在两个数据表之间, 通过主码和外键之间的映射来描述。数据表之间的关系有三种, 即一对一关系、一对多关系和多对多关系。

① 一对一关系 若对表 A 中的每一个记录, 表 B 中只有一个记录与之相联系, 反之也同, 则这两个表之间即是一对一关系。由此可知, 建立一对一关系的两个数据表必须有相同的主码。在实际的数据库设计中, 一对一关系很少遇到。一般只是某一数据表的属性数超过关系数据库的限制, 而将一个表拆分成两个表以保存一个实体的信息时, 这两个数据表之间才建立了一对一关系。

② 一对多关系 若对表 A 中的每一个记录, 表 B 中有零个或多个记录与之相联系, 反之, 对表 B 中的每一个记录, 表 A 中只有一个记录与之相联系, 则这两个表之间是一对多关系。一对一关系是数据库设计中最常遇到的关系, 具有一对多关系的数据表间通过外键建立联系。

③ 多对多关系 若对表 A 中的每一个记录, 表 B 中有零个或多个记录与之相联系, 反

之也同, 则这两个表之间是多对多关系。在关系数据库中多对多关系不应直接存在, 而应通过一个中间表将一个多对多关系转化成多个一对多关系。

1.4 关系运算

对关系数据库进行查询时, 若要找到用户关心的数据, 需要对关系进行一定的关系运算, 关系运算有两种: 一种是传统的集合运算(交、并、差、广义笛卡尔积等); 另一种是专门的关系运算(选择、投影、连接)。

关系运算有一个很明显的特点: 操作对象和运算的结果都是关系。

1.4.1 传统的集合运算

(1) 交运算

关系 R 和关系 S 具有相同的属性列, 且相应的属性取自同一个域, 则关系 R 与关系 S 做交运算, 结果由既属于 R 又属于 S 的所有元组组成。记为: $R \cap S$ 。

(2) 并运算

关系 R 和关系 S 具有相同的属性列, 且相应的属性取自同一个域, 则关系 R 与关系 S 做并运算, 结果由属于 R 或属于 S 的所有元组组成。记为: $R \cup S$ 。

(3) 差运算

关系 R 和关系 S 具有相同的属性列, 且相应的属性取自同一个域, 则关系 R 与关系 S 做差运算, 结果由属于 R 而不属于 S 的所有元组组成。记为: $R - S$ 。

(4) 广义笛卡尔积

两个属性列分别为 n 和 m 的关系 R 和 S , 它们的广义笛卡尔积是一个 $(n+m)$ 列的元组的集合, 元组的前 n 列是关系 R 的一个元组, 后 m 列是关系 S 的一个元组。若 R 有 k_1 个元组, S 有 k_2 个元组, 则关系 R 和关系 S 的广义笛卡尔积有 $k_1 \times k_2$ 个元组。记为: $R \times S$ 。

设关系 R 和关系 S 分别如表 1-2 和表 1-3, 则: $R \cap S$ 的结果如表 1-4 所示, $R \cup S$ 的结果如表 1-5 所示, $R - S$ 的结果如图 1-6 所示, $R \times S$ 的结果如表 1-7 所示。

表 1-2 关系 R

A	B	C
4	3	1
2	5	6

表 1-3 关系 S

A	B	C
13	2	3
2	5	6
2	1	3

表 1-4 $R \cap S$

A	B	C
2	5	6

表 1-5 $R \cup S$

A	B	C
4	3	1
2	5	6
13	2	3
2	1	3

表 1-6 $R - S$

A	B	C
4	3	1

表 1-7 $R \times S$

R.A	R.B	R.C	S.A	S.B	S.C	R.A	R.B	R.C	S.A	S.B	S.C
4	3	1	13	2	3	2	5	6	13	2	3
4	3	1	2	5	6	2	5	6	2	5	6
4	3	1	2	1	3	2	5	6	2	1	3

想一想: $S - R$ 的结果是什么?

1.4.2 关系运算

(1) 选择

选择运算即在关系中选择满足某些条件的元组, 也就是说, 选择运算是在二维表中选择满足指定条件的行。例如, 在学生表(表 1-1)中, 若要找出所有性别是女的学生元组, 就

可以使用选择运算来实现,条件是:性别=女。记为: $\delta_{\text{性别}=\text{“女”}}$ 学生表,如表 1-8 所示:

表 1-8 $\delta_{\text{性别}=\text{“女”}}$ 学生表

学 号	姓 名	性 别	年 龄	系 别
2005002	张小珏	女	20	化工系
2005003	徐菁	女	18	信息系

(2) 投影

投影运算是在关系中选择某些属性列。例如,在学生表中,若要显示所有学生的学号、姓名和性别,那么可以使用投影运算来实现。记为: $\Pi(\text{学号, 姓名, 性别})$ 学生表,如表 1-9 所示。

表 1-9 $\Pi(\text{学号, 姓名, 性别})$ 学生表

学 号	姓 名	性 别	学 号	姓 名	性 别
2005001	李明	男	2005003	徐菁	女
2005002	张小珏	女	2005004	张凡	男

(3) 连接

连接运算是从两个关系(设为 R、S)的笛卡尔积中选取属性间满足一定条件的元组。记为: $R \bowtie_{\text{条件}} S$ 。分为等值连接,不等值连接,自然连接。

假设在表 1-1 “学生表”后,还有一个如表 1-10 所示的“成绩表”,则“学生表 $\bowtie_{\text{成绩表}}$ 成绩表”的结果如表 1-11 的所示。

表 1-10 成绩表

学 号	课程号	成 绩	学 号	课程号	成 绩
2005001	01	85	2005002	03	60
2005001	02	80	2005003	01	82

表 1-11 学生表 $\bowtie_{\text{成绩表}}$ 成绩表

学生表.学号	姓 名	性 别	年 龄	系 别	成绩表.学号	课程号	成 绩
2005001	李明	男	19	机械系	2005001	01	85
2005001	李明	男	19	机械系	2005001	02	80
2005002	张小珏	女	20	化工系	2005002	03	60
2005003	徐菁	女	18	信息	2005003	01	82

想一想: “学生表 $\bowtie_{\text{成绩表}}$ 成绩表”的结果是什么?

自然连接: R 和 S 具有相同的属性组,则 R 和 S 的自然连接的结果为: R 和 S 做等值连接,并在结果中将重复的值去掉。记作: $R \bowtie S$ 。

“学生表 $\bowtie$ 成绩表”的结果如表 1-12 所示。

表 1-12 学生表 $\bowtie$ 成绩表

学 号	姓 名	性 别	年 龄	系 别	课程号	成绩	学 号	姓 名	性 别	年 龄	系 别	课程号	成绩
2005001	李明	男	19	机械系	01	85	2005002	张小珏	女	20	化工系	03	60
2005001	李明	男	19	机械系	02	80	2005003	徐菁	女	18	信息	01	82

1.5 关系规范化

关系模型数据库中要求关系必须规范化,即关系模型中的每一个关系模式均要满足一定的条件。

在以关系模型为基础的关系数据库中,关系用来描述现实世界。如果随意建立关系模型,数据库在应用中将产生很多弊病,如关系中数据的冗余度大大增加,在数据的操作中出现更新异常、插入异常及删除异常等。为了避免和克服这些弊病,关系数据库之父 E.F.Codd 博士于 1971 年提出了规范化理论,即关系模型中的每个关系模式必须满足一定的规范化要求,从而使数据库结构合理,消除存储异常,使数据冗余尽量小,便于数据的插入、删除和更新等。目前这一研究已经取得了很大的进展,先后出现了六种范式,并为每一级定义了相应的约束条件集(范式),从第一范式(1NF)到第五范式。满足最低要求约束的关系模式称为 1NF,在此基础上再进一步满足一些要求的称 2NF,其余依次类推。较高层次的范式比较低层次的范式具有更合乎要求的性质。在实际应用中,一般规范到 3NF。各个范式间的联系如下:

$$5NF \subset 4NF \subset BCNF \subset 3NF \subset 2NF \subset 1NF$$

其中,符号“ $\subset$ ”表示包含。

1.5.1 第一范式(1NF)

1NF 是关系数据库的关系模式应满足的最起码的条件,即关系模式的每个属性都是不可再分的最小数据单位。不属于 1NF 的关系称为非规范化关系。

表 1-13 描述的关系模式 UN_test(学号,姓名,性别,年龄,课程号,成绩,系别和系主任)中,“系别和系主任”不是最小数据项,它可以再分为“系别”和“系主任”两个属性。显然该关系违反了第一范式,应转换为表 1-14 所表示的关系模式 UN(学号,姓名,性别,年龄,课程号,成绩,系别,系主任)。

表 1-13 UN_test 表

学号	姓名	性别	年龄	课程号	成绩	系别和系主任	学号	姓名	性别	年龄	课程号	成绩	系别和系主任
2005001	李明	男	19	01	85	机械系 郑	2005003	徐菁	女	18	01	82	信息系 王
2005001	李明	男	19	02	80	机械系 郑	2005004	张凡	男	21	01	95	信息系 王
2005002	张小珏	女	20	03	60	化工系 张							

表 1-14 UN 表

学号	姓名	性别	年龄	课程号	成绩	系别	系主任	学号	姓名	性别	年龄	课程号	成绩	系别	系主任
2005001	李明	男	19	01	85	机械系	郑	2005003	徐菁	女	18	01	82	信息系	王
2005001	李明	男	19	02	80	机械系	郑	2005004	张凡	男	21	01	95	信息系	王
2005002	张小珏	女	20	03	60	化工系	张								

1.5.2 第二范式(2NF)

如果关系模式符合第一范式,而且不是主码的属性均完全依赖于整个主码,则该关系模式符合第二范式。

表 1-14 所表示的关系模式 UN(学号,姓名,性别,年龄,课程号,成绩,系别,系主任)符合第一范式。但 UN 中存在如下所示的非主码属性不完全依赖于主码的情况,所以该

关系模式不符合第二范式。



在实际应用中，该关系模式将会出现下列问题。

(1) 数据冗余、更新异常

当一名学生选择一门课时，姓名、性别、年龄、系名和系主任就必须重复存储一次，造成数据冗余。如果要更换系主任，就必须逐一修改每条记录。若有疏忽，则会造成数据的不一致性（同一系的学生有不同的系主任）。

(2) 插入异常

如果某个系刚成立，尚无学生，或者有了学生但还没有选课，则造成属性值为空或部分为空，从而导致不能进行插入操作，无法将该系统的系名和系主任插入到 UN 表中。

(3) 删除异常

反过来，如果某个系的学生全部毕业了，删除该系学生及其选课信息的同时，会把系名和系主任的信息同时删掉。

将上述非 2NF 范式的 UN 表关系模式转换为 2NF 范式通常采用模式分解方法，将其分解为以下的两个关系模式，如表 1-15 SG（学号，课程号，成绩）和表 1-16 SDM（学号，姓名，性别，年龄，系别，系主任）的关系模式。SG 表中成绩完全依赖于主码为学号和课程号。SDM 表中符合 2NF 范式。SDM 表中姓名，性别，年龄，系别，系主任都完全依赖于主码学号。这两个关系模式均符合 2NF 范式。

表 1-15 SG 表

学 号	课程号	成 绩	学 号	课程号	成 绩
2005001	01	85	2005003	01	82
2005001	02	80	2005004	01	95
2005002	03	60			

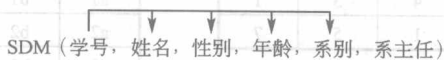
表 1-16 SDM 表

学 号	姓 名	性别	年龄	系 别	系主任	学 号	姓 名	性别	年龄	系 别	系主任
2005001	李明	男	19	机械系	郑	2005003	徐菁	女	18	信息系	王
2005001	李明	男	19	机械系	郑	2005004	张凡	男	21	信息系	王
2005002	张小珏	女	20	化工系	张						

1.5.3 第三范式 (3NF)

如果关系模式符合第二范式，而且所有非主属性相互独立，则该关系模式符合第三范式。如表 1-15 的关系模式就符合 3NF 范式。

表 1-16 SDM（学号，姓名，性别，年龄，系别，系主任）的关系模式还存在以下所示的传递依赖关系：学号决定系别，系别决定系主任。因此 SDM 符合 2NF 范式而不符合 3NF 范式。



此关系仍然存在大量的数据冗余，在插入、删除和更新记录时也将产生类似 UN 表的异常现象，因此有必要对该关系模式进一步规范。通常规范化采用的方法也是模式分解法，