



高职高专“十一五”规划教材



SQL SERVER

2000

SQL Server 2000 数据库

● 胡莹瑾 主编 张鸿彦 李杰 副主编



化学工业出版社



· 北 京 ·

本书通过大量实例讲述了大型关系数据库管理系统 SQL Server 2000 的使用方法, 通俗易懂, 实用性强。

本书主要内容包括关系数据库基础、SQL Server 数据库概述、SQL Server 数据库管理、表的管理和使用、Transact-SQL 语言、视图和存储过程、数据的完整性、数据库的日常维护、SQL Server 2000 数据库的安全管理、应用程序开发实例等。

本书主要适用于高职高专院校的电子商务、计算机及信息管理等专业师生使用, 同时也可供相关专业的从业人员参考使用, 并可作为相关专业的培训教材。

图书在版编目(CIP)数据

SQL Server 2000 数据库 / 胡莹瑾主编. —北京: 化学工业出版社, 2008.1

高职高专“十一五”规划教材

ISBN 978-7-122-01738-3

I. S… II. 胡… 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 200584 号

责任编辑: 李彦玲 于 卉

装帧设计: 于 兵

责任校对: 蒋 宇

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: 北京市兴顺印刷厂

787mm×1092mm 1/16 印张 10½ 字数 250 千字 2008 年 1 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: 18.00 元

版权所有 违者必究

前 言

SQL Server 2000 是微软公司推出的大型关系数据库管理系统,它为用户提供了一个功能强大的客户/服务器平台。由于它具有容易使用,可以很好地保证数据的安全性和完整性,优化系统性能,能够快速地构造、部署和管理数据库应用程序等优点,在各行各业中都得到了非常广泛的应用。

SQL Server 2000 是一门实践性很强的课程,在教学过程中应十分重视实践教学环节,加强学生的实际动手能力。本书在讲解必需的理论知识的基础上,通过大量实例介绍了 SQL Server 2000 的相关技术及其应用,具有通俗易懂,简明易学,突出实用性的特点。

本书共分 10 章,主要介绍了 SQL Server 数据库的管理、使用、日常维护、安全管理、应用程序及开发实例等内容。

本书由在教学一线并具有丰富实践经验的教师共同编写。全书由胡莹瑾任主编,并负责全书的统稿工作。张鸿彦、李杰任副主编。其中第 1 章、第 2 章、第 7 章由张鸿彦、胡莹瑾编写,第 5 章由张鸿彦、李杰编写,第 3 章、第 4 章、第 6 章、第 8 章由胡莹瑾、杨振伟、刘凯编写,第 9 章、第 10 章由吴瑞芝、胡莹瑾编写。

由于软件版本的限制,图中部分的账写成了“帐”,但正文均采用正确的写法“账”,特此声明。

由于编者水平有限,书中不妥之处,敬请广大读者批评指正。

编 者

2008 年 1 月

目 录

第1章 关系数据库基础	1
1.1 关系数据库概述	1
1.1.1 基本概念	1
1.1.2 关系数据库	2
1.1.3 结构化查询语言	3
1.2 关键字	4
1.3 数据库设计步骤	5
1.3.1 信息的转换	5
1.3.2 数据库设计步骤	6
1.3.3 数据库设计规范化	9
1.3.4 数据库设计案例	11
本章小结	13
实训	14
第2章 SQL Server 数据库概述	15
2.1 客户机/服务器体系结构	15
2.1.1 客户机/服务器结构概述	15
2.1.2 数据库的 C/S 结构	15
2.2 SQL Server 2000 数据库概述	17
2.2.1 SQL Server 2000 简介	17
2.2.2 SQL Server 2000 的新特性	17
2.3 SQL Server 2000 的常用工具	18
2.3.1 常用工具简介	18
2.3.2 企业管理器的环境	19
2.4 SQL 查询分析器	20
2.4.1 如何激活 SQL 查询分析器	21
2.4.2 查询分析器内主要窗口介绍	23
2.4.3 SQL 查询分析器工具栏图标介绍	24
2.4.4 使用 SQL 查询分析器	25
2.5 启动和关闭 SQL Server 服务	29
本章小结	30
实训	30

第3章 SQL Server 数据库管理	31
3.1 数据库文件	31
3.2 系统数据库和示例数据库	32
3.2.1 系统数据库	32
3.2.2 示例数据库	32
3.3 创建数据库和事务日志	33
3.4 管理数据库	36
3.4.1 扩充数据库文件的长度	36
3.4.2 收缩数据库	36
3.4.3 修改数据库选项	38
3.4.4 修改数据库的名称	38
3.4.5 删除数据库	39
本章小结	39
实训	39
第4章 表的管理和使用	40
4.1 数据类型	40
4.1.1 数字数据类型	40
4.1.2 货币数据类型	40
4.1.3 日期/时间数据类型	41
4.1.4 字符数据类型	41
4.1.5 二进制数据类型	41
4.1.6 Unicode 数据类型	41
4.1.7 其它数据类型	42
4.2 创建和管理表	42
4.2.1 创建表	42
4.2.2 重新命名表	45
4.2.3 修改表	45
4.2.4 删除表	45
4.3 索引	46
4.3.1 索引概述	46
4.3.2 创建和删除索引	47
本章小结	49
实训	49
第5章 Transact-SQL 语言	51
5.1 Transact-SQL 概述	51
5.2 SELECT 语句的基本结构	51
5.3 简单查询	51
5.3.1 使用 SELECT 子句选取子段	52

1E	5.3.2 使用 INTO 子句	55
1E	5.3.3 使用 WHERE 子句	56
CE	5.3.4 使用 ORDER BY 子句	60
SE	5.4 统计	61
CE	5.4.1 使用集合函数	61
EE	5.4.2 使用 GROUP BY 子句	62
DE	5.4.3 使用 COMPUTE 子句	63
DE	5.5 多表查询	64
DE	5.5.1 使用交叉连接	64
DE	5.5.2 使用内连接	65
DE	5.5.3 使用外连接	66
DE	5.5.4 使用自连接	69
DE	5.5.5 合并结果集	69
DE	5.6 子查询	70
	5.6.1 使用子查询进行比较测试	70
04	5.6.2 使用子查询进行集成测试	71
04	5.6.3 使用子查询进行存在性测试	71
04	5.6.4 使用子查询进行批量比较测试	72
04	5.7 添加数据	74
14	5.7.1 使用 INSERT...VALUES 语句添加记录	74
14	5.7.2 使用 INSERT...SELECT 语句添加多条记录	75
1E	5.8 修改数据	76
1E	5.9 删除数据	77
SA	5.9.1 使用 DELETE 语句删除记录	77
SA	5.9.2 使用 TRUNCATE TABLE 语句删除所有记录	77
SE	5.10 使用企业管理器查询与更新记录	78
2E	本章小结	81
2E	实训	82
24	第 6 章 视图和存储过程	83
04	6.1 视图	83
74	6.1.1 视图的概念	83
04	6.1.2 使用视图的优点和作用	83
04	6.1.3 创建视图	83
	6.1.4 修改视图的定义	86
1E	6.1.5 查看视图的返回结果	86
1E	6.1.6 显示视图的定义信息	88
1E	6.1.7 重命名和删除视图	89
1E	6.2 存储过程	90
CE	6.2.1 创建和执行存储过程	90

6.2.2 查看和修改存储过程	92
6.2.3 重命名和删除存储过程	94
本章小结	95
实训	96
第7章 数据的完整性	97
7.1 数据完整性概述	97
7.1.1 实体完整性	97
7.1.2 域完整性	97
7.1.3 参照完整性	97
7.1.4 用户定义的完整性	98
7.1.5 维护数据完整性的对策	98
7.2 创建和使用约束	98
7.2.1 主键约束	98
7.2.2 外键约束	100
7.2.3 唯一性约束	102
7.2.4 检查约束	103
7.2.5 缺省约束	104
7.2.6 查看约束	105
7.2.7 删除约束	106
7.3 规则	106
7.3.1 创建规则	107
7.3.2 查看规则	108
7.3.3 规则的绑定与松绑	108
7.3.4 删除规则	111
本章小结	112
实训	112
第8章 数据库的日常维护	113
8.1 数据库的备份和还原	113
8.1.1 使用企业管理器进行数据库的备份	113
8.1.2 使用企业管理器还原数据库	114
8.2 分离与附加数据库	117
8.3 数据的导入与导出	119
本章小结	124
实训	124
第9章 SQL Server2000 数据库的安全管理	125
9.1 注册服务器	125
9.1.1 注册服务器方法	125

9.1.2	删除服务器注册	128
9.1.3	修改注册服务器	128
9.2	SQL Server 登录模式	129
9.2.1	SQL Server 登录认证	129
9.2.2	SQL Server 认证模式的设置	129
9.2.3	登录账户管理	130
9.3	角色	135
9.3.1	服务器角色	135
9.3.2	数据库角色	136
9.4	权限	140
9.4.1	权限的种类	140
9.4.2	权限的设置	141
	本章小结	144
	实训	144
	第 10 章 应用程序开发实例	145
10.1	创建 ODBC 数据源	145
10.2	ADO 对象访问数据库	147
10.3	使用 Visual Basic 开发 SQL Server 应用程序的简单实例	148
	本章小结	154
	实训	154
	附表 1 xsdate 数据库 class 表的全部数据	155
	附表 2-1 xsdate 数据库 student 表的全部数据	156
	附表 2-2 xsdate 数据库 student 表的全部数据续表	157
	附表 3 xsdate 数据库 grade 表的全部数据	158
	参考文献	159

第 1 章 关系数据库基础

学习目标

- 理解关系数据库的基本概念
- 掌握关键字和外关键字的基本定义
- 掌握实体与实体间的联系方式
- 掌握数据库设计方法
- 灵活运用关系规范化的理论对数据库进行优化

1.1 关系数据库概述

1.1.1 基本概念

1. 数据

所谓数据 (Data) 就是描述事物的符号。在我们的日常生活中, 数据无所不在, 数字、文字、图表、图像、声音等都是数据。人们通过数据来认识世界、交流信息。

2. 数据库

数据库 (Database, DB) 就是数据存放的地方。在计算机中, 数据库是与特定主题或用途相关的数据和数据库对象的集合。所谓数据库对象是指表 (Table)、视图 (View)、存储过程 (Stored Procedure)、触发器 (Trigger) 等。数据库存储在文件中。

3. 数据库管理系统

数据库管理系统 (Database Management System, DBMS) 是用于管理数据的计算机系统软件。数据库管理系统使用户能方便地定义和操纵数据, 维护数据的安全性和完整性, 以及进行多用户下的并发控制和恢复数据库。

数据库管理系统提供了用户与数据库之间的软件界面, 可以让用户创建、管理和维护数据库。数据库管理系统通常具有数据定义、数据安全和数据安全等方面的功能。

数据库管理系统可以分为桌面数据库管理系统和客户机/服务器数据库管理系统。桌面数据库管理系统通常用于管理本地机上的小型数据库, 数据量往往不太大, 例如 Access 和 FoxPro 都属于桌面数据库管理系统。客户机/服务器数据库管理系统也简称为数据库服务器, 通常用于客户机/服务器结构、多层结构以及 Web 应用程序中, 并扮演着后端数据库的角色。

本书介绍的 SQL Server 2000 就是一种常用的客户机/服务器数据库管理系统。除此之外, 比较常用的客户机/服务器数据库管理系统还有 Oracle、Sybase 以及 Informix 等。

4. 数据库系统

数据库系统 (Database System, DBS) 狭义地讲是由数据库管理系统、数据库和用户构成, 广义地讲是由计算机硬件、操作系统、数据库管理系统以及在其支持下建立起来的数据库、应用程序、用户和维护人员组成的一个整体。

1.1.2 关系数据库

数据库这一概念提出后,先后出现了几种数据模型。其中基本的数据模型有三种:层次模型、网状模型和关系模型。20世纪60年代末期提出的关系模型具有数据结构简单灵活、易学易懂且具有雄厚的数学基础等特点,从70年代开始流行,发展到现在已成为数据库的标准。目前广泛使用的数据库软件都是基于关系模型的关系数据库管理系统。

1. 关系模型

关系模型(Relational Model, RM)把世界看做是由实体(Entity)和联系(Relationship)构成的。

所谓实体就是指现实世界中具有区分于其它事物的特征或属性并与其它实体有联系的对象。例如课程、学生、班级等都是实体。实体表示的是一类事物,其中的一个具体事物称之为该实体的一个实例,如学生“王静”是学生实体的一个实例。

所谓联系就是指实体之间的关系,即实体之间的对应关系。联系可以分为三种。

(1) 一对一联系(1:1)

若对于实体集 A 中每一个实体,实体集 B 中至多只有一个实体与之联系,反之与实体集 B 中每一个实体,实体集 A 中也至多只有一个实体与之联系。这称为实体集 A 与实体集 B 之间具有一对一联系,记为1:1。

例如学院和院长的联系,假如每个学院只有一个院长,而一个人只能在一个学院任职,则它们是一对一的联系。

(2) 一对多联系(1:N)

若对于实体集 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。而对于实体集 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 有一对多的联系,记为1:N。

例如一个系可负责多门课程的讲授,而一门课只能由一个系的老师来讲,则系与课程是一对多的联系。

(3) 多对多联系(M:N)

若对于实体集 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,反过来对于实体集 B 中的每一个实体,实体集 A 中有 m 个实体($m \geq 0$)与之联系,则称实体集 A 与实体集 B 之间有多对多联系,记为M:N。

例如一个学生可以学习多门课程,一门课程也可有多个学生学习,则学生与课程之间是多对多的联系。

通过联系就可以用一个实体的信息来查找另一个实体的信息。

说明:实体集与实体集间的联系并不是固定不变的,它与系统环境、研究的问题有关,某些问题与前提条件有关。例如在一个学校中,每个老师最多担任一门课主讲,但有些课程有多位主讲老师,则课程和老师问题是一对多联系。在一个学校中,如果有的老师要担任多门课主讲,且有一些课程有多位主讲,则课程和老师之间是多对多联系。在一个学校中如果有的教师担任多门课主讲,而所有课程均只有一位主讲,则课程和老师间是多对一的联系。

2. 关系数据库

所谓关系数据库(Relational Database, RDB)就是基于关系模型的数据库。关系数据

库管理系统（DBMS）就是管理关系数据库的计算机系统软件。

在关系数据库中，所有的数据都是以表格的形式进行组织和存储。每个表格都由列和行构成。列用来描述数据的属性，行代表唯一的一条记录，每一列（也称字段）表示记录中的一个元素。在同一个表中，每一列的名称必须是唯一的，每一条记录必须是唯一的，即不能出现完全相同的两条记录。在不同的表中可以使用相同的列名。

如图 1-1 所示，课程表 course，由若干行和 5 列构成，5 列分别为课程编码 couno、课程名 couname、分类 kind、教师 teacher 和系部编码 departno。每一行代表一门课程的记录，该记录中的每个元素由相关的列来标识。如表中的第一条记录，由 5 列构成，分别是课程编码 001，课程名电子商务概论、分类电子商务、教师丛云茹、系部编码 01。

图 1-1 表“course”中的数据，位置是“xsdate”中、“(local)”上

couno	couname	kind	teacher	departno
001	电子商务概论	电子商务	丛云茹	01
002	网络营销	电子商务	于涛	01
003	电子商务与现代	电子商务	杨振伟	01
004	电子商务网站建设	电子商务	胡莹瑾	01
005	商务谈判	电子商务	狄娜	01
006	物流管理概论	物流管理	李辉	01
007	仓储与运输	物流管理	武慧芳	01
008	物流配送中心设计	物流管理	胡月阳	01
009	遗传学	生物工程	崔海明	02
010	生物统计	生物工程	孙友	02
011	组织培养	生物工程	黄先石	02
012	细胞工程	生物工程	李明远	02
013	SQL 数据库	信息技术	赵明艳	03
014	ASP 程序设计	信息技术	崔俊生	03
015	计算机组成原理	信息技术	杜小明	03
016	计算机网络与通信	信息技术	邹志生	03
017	平面设计	信息技术	王海月	03
018	酒店管理	管理	王月	04
019	俄语导游	管理	李忆	04
020	森林旅游	管理	张国伟	04

图 1-1 课程表信息

在不同的表之间可以建立起关联，通过查询可以使用一个表中的数据来查找其它表中的相关数据。

1.1.3 结构化查询语言

结构化查询语言（Structured Query Language，SQL）是一种应用广泛的数据库语言，用于定义和管理关系数据库中的对象，也用于检索、修改和删除数据库中的数据。

SQL 是一种十分优雅的数据库语言，它已经成为与数据库“交谈”的最有效方法。SQL 提供了一种独特的交互查询方法。在程序中使用 SQL 语言来提出各种问题，这种问题也称为查询（Query）。数据库服务器通过返回满足要求的记录行来“回答”提问。

例如，如果要从一个名为 Books 的表中检索有关 SQL Server 2000 的图书并返回其书号（ISBN）、书名（Title）、作者姓名（AuthorName）以及库存量（Stocks）等信息，可以通过下面的 SQL 查询语句来实现：

```
SELECT ISBN, Title, AuthorName, Stocks FROM Books  
WHERE Title LIKE 'SQL Server 2000'
```

但是，SQL 只是一种数据库语言，其主要功能是访问、查询、更新和管理关系数据库

中的数据。SQL Server 等网络数据库管理系统主要用于存储数据、管理数据和提供数据，以满足客户端连接数据库和存取数据的需要。SQL 本身并不是一种功能完善的程序设计语言，它不能用于构建输入/输出界面、设计数据报表和建立菜单系统。如果要设计数据库应用系统的图形用户界面，应当选择 Visual BASIC、Visual C++或 Delphi 等作为客户端开发工具。

国际标准化组织 (ISO) 和美国国家标准学会 (ANSI) 针对 SQL 制定了一些标准，SQL 已成为大多数关系数据库支持的语言。不过，各数据库厂商往往会在其数据库产品中对 SQL 语言进行某些扩展和修改。在 SQL Server 2000 中使用的 SQL 语言称为 Transact-SQL。

1.2 关 键 字

关键字 (Key) 是关系模型中的一个重要概念。关键字是用来唯一标识表中每一行的属性或属性的集合，又称为关键码、码或键。

1. 候选关键字

如果一个属性集能唯一地标识表的一行而又不含多余的属性，那么这个属性集称为候选关键字 (Candidate Key)。如在表 1-1 中，“系部编码”和“系部名称”均是关系“系部”的候选关键字。

候选关键字有如下性质。

① 在一个关系中，候选关键字可以有多个。例如表 1-1 的系关系中，“系部编码”、“系部名称”都是候选关键字。

表 1-1 系部数据表

系部编码	系 部 名 称	系部编码	系 部 名 称
01	管理系	03	机电系
02	生物工程系	04	旅游系

② 关键字可以由一个属性构成，也可以由多个属性构成。如表 1-2 中，“学号+课程号”是候选关键字。关键字不可能再与其它的属性构成新的候选关键字。

表 1-2 成绩表

学 号	课程号	成 绩	学 号	课程号	成 绩
00000001	001	75	00000002	003	80
00000001	002	85	00000003	001	75
00000001	003	80	00000003	003	90
00000002	002	70			

2. 主关键字

主关键字 (Primary Key) 是从候选关键字中挑选出来，作表的行的唯一标识。一个表只有一个主关键字。主关键字又可以称为主键。如表 1-1 中，“系部编码”是主关键字。

3. 公共关键字

在关系数据库中，如果两个关系中具有相同的属性或属性集合，那么这个属性或属性集合被称为这两个关系的公共关键字 (Common Key)。如在表 1-3 和表 1-1 中都有“系部编码”字段，则“系部编码”是这两个表的公共关键字。

表 1-3 班级表

班 级 编 码	系部编码	班 级 名 称	班 级 编 码	系部编码	班 级 名 称
20040001	01	04 电子商务	20040007	03	04 网络技术
20040004	02	04 生物技术	20040009	04	04 酒店管理

4. 外关键字

如果公共关键字在一个关系中是主关键字,那么这个公共关键字被称为另一个关系的外关键字 (Foreign Key)。由此可见,外关键字表示了两个关系之间的联系。以另一个关系的外关键字作主关键字的表被称为主表,具有此外关键字的表被称为主表的从表。外关键字又称作外键。如在表 1-1 中,“系部编码”是主关键字,则它是表 1-3 的外关键字。

1.3 数据库设计步骤

如何根据企业中用户的需求及企业生存环境,在指定的数据库管理系统上,设计企业数据库的逻辑模型,最后建成企业数据库。这是一个从现实世界向计算机数据世界转换的过程。在设计关系数据库时,最重要的工作就是妥善规划数据表的配置,以避免产生数据重复、数据不一致或数据表间的关联不完整等问题。

关系数据库的设计工作大致可以分为两个阶段。

第一阶段:收集完整、必要的数项目,并转换成数据表的属性形式。

第二阶段:将收集的属性做适当分类后,归入不同的数据表中,并建立数据表间的联系。

1.3.1 信息的转换

信息是对现实世界客观存在的事物的反映,数据则是用来表示信息的一种符号。反映客观事物状态的数据,经过一定的组织,成为计算机内部的数据,将经历三个不同的状态:现实世界、信息世界 (概念世界)、计算机世界 (数据世界),如图 1-2 所示。

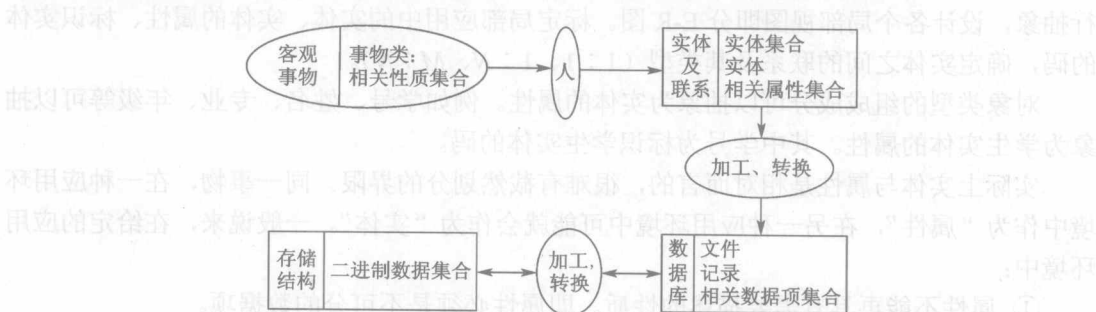


图 1-2 三个不同的世界

在不同的世界中使用的概念与术语是不同的,但它们在转换过程中都有一一对应的关系,如表 1-4 所示。

例如现实世界中的一个“事物”,对应于信息世界中的一个“实体”。实体可以是一个学生,一个零件或一张订货合同。事物总是有一些性质反映事物的特征。因此实体总是有

表 1-4 三个不同世界术语对照表

客 观 世 界	信息世界 (概念世界)	数 据 世 界
组织 (事物及其联系)	实体及其联系 (概念模型)	数据库 (数据模型)
事物类 (总体)	实体集	文件
事物 (对象、个体)	实体	记录
特征 (性质)	属性	数据项

一些属性,反映实体的特征,如学生的学号、姓名等。实体的属性在计算机世界中用数据项描述,实体属性的集合在计算机世界中用记录描述。具有相同属性的事物的集合,如一群学生、一群教师,就形成了事物类,它们是信息世界中的实体集(简称实体),在计算机世界中,则形成一个个数据文件,如学生文件、教师文件。但是客观事物是复杂的,涉及同一事务的事物有多个,相互之间又有错综复杂的联系,因此反映在信息世界就有实体及它们的联系,反映在计算机世界就形成了逻辑数据库(许多逻辑数据文件的集合)。

1.3.2 数据库设计步骤

1. 概念结构设计

(1) 对现实世界进行需求分析

对现实世界要处理的对象(组织、部门企业等)进行详细调查。调查的重点是“数据”和“处理”,通过调查获得每个信息使用者对数据库的要求。

(2) 建立信息世界中 E-R 模型

实体-联系模型(Entity-Relationship Model)是 P.P.S.Chen 于 1976 年提出的一种概念模型,用 E-R 图来描述一个系统中的数据及其之间关系。在 E-R 图中,用长方形表示实体集,在长方形框内写上实体名。用菱形表示实体间联系,菱形框内写上联系名。用无向边把菱形和有关实体相连接,在无向边旁标上联系的类型,如 1 或 M 或 N 。用椭圆形表示实体或联系的属性并以无向边相连。

概念模型的建立分为两步。

第一步,首先要根据需求分析的结果(数据流图、数据字典等)对现实世界的数据进行抽象,设计各个局部视图即分 E-R 图。标定局部应用中的实体、实体的属性、标识实体的码,确定实体之间的联系及其类型(1:1、1:N、 $M:N$)。

对象类型的组成成分可以抽象为实体的属性。例如学号、姓名、专业、年级等可以抽象为学生实体的属性。其中学号为标识学生实体的码。

实际上实体与属性是相对而言的,很难有截然划分的界限。同一事物,在一种应用环境中作为“属性”,在另一种应用环境中可能就会作为“实体”。一般说来,在给定的应用环境中:

① 属性不能再具有需要描述的性质。即属性必须是不可分的数据项。

② 属性不能与其它实体具有联系。联系只发生在实体之间。

例:学籍管理局部应用中主要涉及的实体包括学生、宿舍、档案材料、班级、班主任。那么,这些实体之间的联系又是怎样的呢?

由于一个宿舍可以住多个学生,而一个学生只能住在某一个宿舍中,因此宿舍与学生之间是 1:n 的联系。由于一个班级往往有若干名学生,而一个学生只能属于一个班级,因此班级与学生之间也是 1:n 的联系。由于班主任同时还要教课,因此班主任与学生之间存

在指导联系，一个班主任要教多名学生，而一个学生只对应一个班主任，因此班主任与学生之间也是 $1:n$ 的联系。而学生和他自己的档案材料之间，班级与班主任之间都是 $1:1$ 的联系。

接下来我们需要进一步斟酌该 E-R 图，做适当调整。

在一般情况下，性别通常作为学生实体的属性，但在本局部应用中，由于宿舍分配与学生性别有关，应该把“性别”作为实体对待。

这样，得到学籍管理局局部应用的分 E-R 图，如图 1-3 所示。

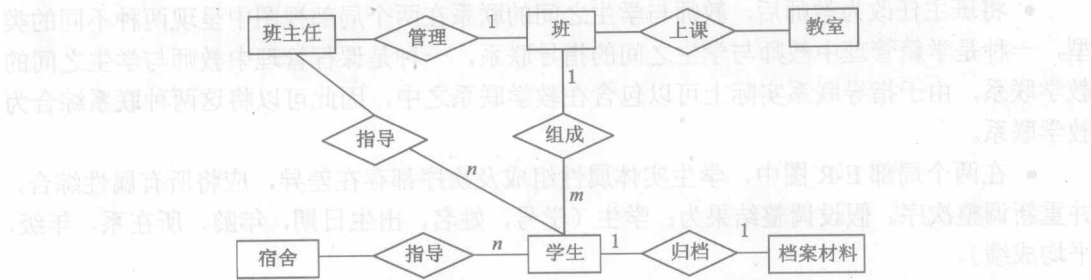


图 1-3 学籍管理局局部应用的分 E-R 图

因篇幅有限，该 E-R 图中省略了各个实体的属性描述。这些实体的属性分别为：

学生（学号，姓名，出生日期）

档案材料（档案号，.....）

班级（班级号，学生人数）

班主任（职工号，姓名，性别，是否为优秀班主任）

宿舍（宿舍编号，地址，人数）

教室（教室编号，地址，容量）

其中有下列划线的属性为实体的码。

同样的方法，我们可以得到课程管理局局部应用的分 E-R 图，如图 1-4 所示。

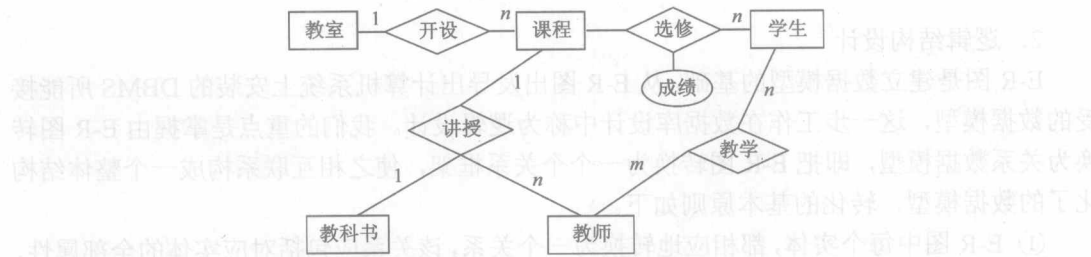


图 1-4 课程管理局局部应用的分 E-R 图

各实体的属性分别为：

学生（姓名，学号，性别，年龄，所在系，年级，平均成绩）

课程（课程号，课程名，学分）

教师（职工号，姓名，性别，职称）

教科书（书号，书名，定价）

教室（教室编号，地址，容量）

第二步，集成分 E-R 图，设计全局 E-R 图。

下面我们来看看如何生成学校管理系统的初步 E-R 图。我们着重介绍学籍管理局部视图与课程管理局部视图的合并。这两个分 E-R 图存在着多方面的冲突。

- 班主任实际上也属于教师，也就是说学籍管理中的班主任实体与课程管理中的教师实体在一定程度上属于异名同义，可以将学籍管理中的班主任实体与课程管理中的教师实体统一称为教师，统一后教师实体的属性构成为：教师（职工号，姓名，性别，职称，是否为班主任）。

- 将班主任改为教师后，教师与学生之间的联系在两个局部视图中呈现两种不同的类型，一种是学籍管理中教师与学生之间的指导联系，一种是课程管理中教师与学生之间的教学联系，由于指导联系实际上可以包含在教学联系之中，因此可以将这两种联系综合为教学联系。

- 在两个局部 E-R 图中，学生实体属性组成及次序都存在差异，应将所有属性综合，并重新调整次序。假设调整结果为：学生（学号，姓名，出生日期，年龄，所在系，年级，平均成绩）。

解决上述冲突后，学籍管理分 E-R 图与课程管理分 E-R 图合并为初步 E-R 图，如图 1-5 所示。



图 1-5 学生管理子系统基本 E-R 图

2. 逻辑结构设计

E-R 图是建立数据模型的基础，从 E-R 图出发导出计算机系统上安装的 DBMS 所能接受的数据模型，这一步工作在数据库设计中称为逻辑设计。我们的重点是掌握由 E-R 图转换为关系数据模型，即把 E-R 图转换为一个个关系框架，使之相互联系构成一个整体结构化了的数据模型，转化的基本原则如下。

① E-R 图中每个实体，都相应地转换为一个关系，该关系应包括对应实体的全部属性，并应根据该关系表达的语义确定出关键字，因为关系中的关键字属性是实现不同关系联系的主要手段。

② 一个 $m:n$ 联系转换为一个关系模式。该关系的关键字由两个实体的关键字组合在一起成为组合关键字，并附上联系本身的属性。

③ 一个 $1:n$ 联系可以转换为一个独立的关系模式，也可以与 n 端对应的关系模式合并。如果转换为一个独立的关系模式，则与该联系相连的各实体的码以及联系本身的属性均转换为关系的属性，而关系的码为 n 端实体的码；也可将 1 方实体的主关键字加入到多方中，作为多方的一个属性。