

# SQL Server 2012 数据库原理及应用

刘娜 董引娣 周健飞 ◇ 主编

SQL SERVER 2012 SHUJUKU YUANLI JI YINGYONG

HEUP 哈尔滨工程大学出版社

# SQL Server 2012 数据库原理及应用

周健飞 ◇ 主 编

郝淑娟 ◇ 副主编



HEUP 哈尔滨工程大学出版社

## 内 容 简 介

本书讲授数据库基本原理和技术,并以 SQL Server 2012 为平台介绍数据库管理系统的应用,由浅入深、循序渐进地介绍各个知识点,提供了大量例题并做了深入剖析,有助于读者理解概念和巩固知识。

本书可以作为各类院校计算机科学与技术及相关专业“数据库原理与应用”课程的教学用书。对于计算机应用人员和计算机爱好者而言,本书也是一本实用的自学参考书。

## 图书在版编目(CIP)数据

SQL Server 2012 数据库原理及应用/刘娜,董引娣,周健飞主编. —哈尔滨:哈尔滨工程大学出版社, 2017.6

ISBN 978 - 7 - 5661 - 1547 - 8

I. ①S… II. ①刘… ②董… ③周… III. ①关系数据库系统 IV. ①TP311.138

中国版本图书馆 CIP 数据核字(2017)第 137428 号

选题策划 吴振雷  
责任编辑 张忠远 周一瞳  
封面设计 博鑫设计

---

出版发行 哈尔滨工程大学出版社  
社 址 哈尔滨市南岗区东大直街 124 号  
邮政编码 150001  
发行电话 0451 - 82519328  
传 真 0451 - 82519699  
经 销 新华书店  
印 刷 黑龙江龙江传媒有限责任公司  
开 本 787 mm × 1 092 mm 1/16  
印 张 15  
字 数 398 千字  
版 次 2017 年 6 月第 1 版  
印 次 2017 年 6 月第 1 次印刷  
定 价 39.80 元

<http://www.hrbeupress.com>

E-mail: [heupress@hrbeu.edu.cn](mailto:heupress@hrbeu.edu.cn)

---

# 前 言

随着我国改革开放的进一步深化,高等教育也得到了快速发展。各地高校紧密结合地方经济建设发展需要,科学运用市场调节机制,加大了使用信息科学等现代科学技术提升、改造传统学科专业的投入力度,通过教育改革合理调整和配置了教育资源,优化了传统学科专业,积极为地方经济建设输送人才,为我国经济社会的快速、健康和可持续发展以及高等教育自身的改革发展做出了巨大贡献。但是,高等教育质量还需要进一步提高以适应经济社会发展的需要,不少高校的专业设置和结构不尽合理,教师队伍整体素质亟待提高,人才培养模式、教学内容和方法需要进一步转变,学生的实践能力和创新精神还需加强。

数据库技术是计算机技术领域中发展最快的技术之一,也是应用最为广泛的技术之一,它已经成为计算机信息系统的核心技术和重要基础。微软公司在 Windows 系列平台上开发的 SQL Server 是一个功能完备的数据库管理系统,一经推出就以其易用性得到了很多用户的青睐,它使用 Transact - SQL 语言在客户机与服务器之间发送请求。SQL Server 2012 是微软公司于 2011 年继 SQL Server 2008 之后发布的版本。从 SQL Server 2008 到 SQL Server 2012,不仅是数据库系统具有更高的性能、更强的处理能力,新版本的系统还带来了许多新的、在旧版本中从未出现过的特性。SQL Server 2012 作为已经为云技术做好准备的信息平台,能够快速构建相应的快速解决方案,实现本地和公有云之间的数据扩展。

本书由重庆工业职业技术学院刘娜老师担任第一主编并进行全书的统编,负责编写第五章和第八章,共计 7.04 万字;重庆城市管理职业学院董引娣老师担任第二主编,负责编写第六章、第七章和第九章,共计 8.19 万字;重庆工业职业技术学院周健飞老师担任第三主编,负责编写第十章和第十一章,共计 7.04 万字;大庆师范学院林芳老师负责编写第三章,共计 5.4 万字;大庆师范学院邱忠阳负责编写第一章、第二章的 2.1 节及 2.2 部分内容,共计 5.04 万字;大庆师范学院郝淑娟负责编写第二章的 2.2 节部分内容及第四章,共计 5.06 万字。

为了满足当前高职高专院校人才培养和当今社会对人才需求的要求,很多学校的相关专业均开设了有关数据库技术的课程,而在众多的数据库系统中,SQL Server 以其兼具对大型数据库技术的要求和易于实现等特点被许多院校列为必修课程。本书正是结合这一实际需要以及最新的数据库技术知识而编写的。

本书内容由浅入深、循序渐进、通俗易懂,非常适合自学,既讲授一般性的数据库原理,又突出数据库实际应用技术。书中提供了大量例题,有助于读者理解概念、巩固知识、掌握要点、攻克难点。本书可以作为各类院校计算机科学与技术及相关专业的“数据库原理与技术”课程的教学用书,对于计算机应用人员和计算机爱好者而言也是一本实用的自学参考书。

编 者  
2017 年 1 月

# 目 录

<b>第 1 章 数据库概述</b> .....	1
1.1 数据库的基本概念及数据库系统 .....	1
1.2 数据模型概述 .....	3
1.3 关系数据库的基本原理 .....	9
1.4 SQL Server 概述 .....	18
<b>第 2 章 SQL Server 2012 的安装及管理工具</b> .....	23
2.1 选择 SQL Server 2012 数据库版本 .....	23
2.2 SQL Server Management Studio 管理工具 .....	35
<b>第 3 章 SQL Server 2012 数据库对象的操作</b> .....	41
3.1 数据库的基本操作 .....	41
3.2 数据库表的基本操作 .....	59
<b>第 4 章 数据库标准语言 SQL</b> .....	74
4.1 SQL 语言 .....	74
4.2 数据查询 .....	78
4.3 数据更新 .....	91
4.4 数据控制 .....	93
<b>第 5 章 存储过程及触发器</b> .....	96
5.1 存储过程概述 .....	96
5.2 存储过程的操作 .....	97
5.3 存储过程的管理 .....	103
5.4 触发器 .....	106
5.5 触发器的创建 .....	108
5.6 触发器的管理 .....	115
<b>第 6 章 视图和函数</b> .....	119
6.1 视图 .....	119
6.2 自定义函数 .....	125
<b>第 7 章 数据完整性和规则</b> .....	130
7.1 约束 .....	130
7.2 默认值 .....	142
7.3 规则 .....	144



<b>第8章 索引和游标的使用</b>	146
8.1 索引概述	146
8.2 索引的应用	149
8.3 游标	156
<b>第9章 SQL Server 的安全管理</b>	166
9.1 SQL Server 的安全模型	166
9.2 服务器的安全性	168
9.3 数据库的安全性	177
9.4 数据库用户角色	180
9.5 权限	183
<b>第10章 数据库设计的步骤</b>	188
10.1 数据库设计概述	188
10.2 需求分析	190
10.3 概念结构设计	197
10.4 逻辑结构设计	206
10.5 数据库实施	210
10.6 数据库运行与维护	211
<b>第11章 数据库应用程序开发</b>	213
11.1 数据库应用程序设计方法	213
11.2 数据库应用程序的体系结构	214
11.3 数据库与应用程序的接口	220
11.4 数据库应用程序开发	230
<b>参考文献</b>	233

# 第1章 数据库概述

数据库技术是信息社会的重要基础技术之一,是计算机科学技术领域中发展最为迅速的重要分支。数据库技术是一门综合性技术,涉及操作系统、数据结构、算法设计、程序设计等基础理论知识,因此,在计算机科学中是将其作为专门的学科进行学习和研究的,并为之指导和推动应用。对普通计算机用户而言,虽更多注重于学习数据库技术的实际应用方法,但学习、掌握一些必需的、实用的基础知识也是非常重要的。对数据库技术的应用,特别是在开发应用系统时尤为重要。因此,本章将以一定篇幅介绍数据库技术相关基础知识,使读者在学习、应用数据库技术的过程中,做到既知其然又知其所以然。

## 1.1 数据库的基本概念及数据库系统

数据、数据库、数据库管理系统和数据库系统是四个密切相关的基本概念。

### 1.1.1 数据库的基本概念

我们先简单地介绍一下数据库的几个基本概念。

#### 1. 数据

数据(Data)是描述客观事物的符号记录,可以是数字、文字、图形、图像、声音、语言等,经过数字化后存入计算机。事物可以是可触及的对象(一个人、一棵树、一个零件等),可以是抽象事件(一次球赛、一次演出等),也可以是事物之间的联系(一张借书卡、一张订货单等)。

数据和关于数据的解释是不可分的。数据的形式本身并不能完全表达其内容,了解的人 would 知道一组数据的具体含义,但是,不知道的人无法理解。因此,数据应该由数据值及其含义两部分组成,也就是需要经过语义解释。数据与其语义是密不可分的。

#### 2. 数据库

数据库(Database, DB)是存放数据的仓库,是长期存储在计算机内的、有组织的、可共享的数据集合。在数据库中集中存放了一个有组织的、完整的、有价值的数据库资源,如图书管理、学生管理、人事管理、物业管理等。它可以供各种用户共享,有最小冗余度、较高的数据库独立性和易扩展性。

#### 3. 数据库管理系统

数据库管理系统(Database Management System, DBMS)是指位于用户与操作系统之间的一层数据管理软件。数据库在建立、运行和维护时由数据库管理系统统一管理、统一控制。数据库管理系统是一组计算机程序,使用户能方便地定义数据和操纵数据,并能够保证数据的安全性和完整性、多用户对数据的并发使用及发生故障后的系统恢复。用户使用数据库是有目的的,数据库管理系统是帮助用户达到这一目的的工具和手段。

#### 4. 数据库系统

数据库系统(Database System, DBS)是指在计算机系统中引入数据库后的系统构成,一



般由数据、数据库管理系统(及其开发工具)、应用系统、数据库管理员和用户构成。

### 5. 数据库系统管理员

数据库系统管理员(Database Administrator, DBA)是负责数据库的建立、使用和维护的专门人员。

## 1.1.2 数据库系统的组成

数据库系统包括数据、硬件、软件 and 用户四部分。

### 1. 数据

数据是构成数据库的主体,是数据库系统的管理对象。

### 2. 硬件

硬件是数据库系统的物理支撑,包括 CPU、内存、外存及 I/O 设备等。

### 3. 软件

软件包括系统软件和应用软件。系统软件包括操作系统和数据库管理系统,数据库管理系统是数据库系统中最重要的核心软件;应用软件是在数据库管理系统的支持下由用户根据实际需求开发的应用程序。

### 4. 用户

用户包括专业用户、非专业用户和数据库管理员:专业用户是指应用程序员,负责设计和编制应用程序,通过应用程序存取和维护数据库,为最终用户准备应用程序;非专业用户即最终用户,一般是指非计算机专业人员,他们通过应用系统提供的用户接口界面以交互式操作使用数据库,交互式操作通常为菜单驱动、图形显示、表格操作等;数据库管理员全面负责数据库系统的管理、维护和正常使用,保持数据库始终处于最佳工作状态,对于大型数据库系统,要求配置专门的 DBA。数据库管理员的主要职责如下:

- ①参与数据库设计的全过程;
- ②定义数据库的安全性和完整性约束条件;
- ③决定数据库的存储和读取策略;
- ④监督控制数据库的使用和运行并及时处理运行程序中出现的問題;
- ⑤改进数据库系统和重组数据库。

## 1.1.3 数据库系统的特点

### 1. 数据结构化

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。整个数据库按一定的结构形式构成,数据在记录内部和记录类型之间相互关联,用户可以通过不同的路径存取数据。数据库系统主要实现整体数据的结构化。

### 2. 数据的共享性高、冗余度低、易扩充

数据库系统的数据面向整个系统,所以可以为多用户、多应用共享。每个用户只与数据库中的一部分数据发生联系,用户数据可以重叠,多个用户可以同时存取数据而互不影响,因此,大大提高了数据库的使用效率。数据共享可以大大减少冗余度、节约存储空间;数据共享还能避免数据之间的不一致性,这种数据的不一致性是指同一数据在每次复制时的值不一样;数据共享还能使数据库系统具有弹性大、易扩充的特点。



### 3. 数据独立性高

数据独立性主要从物理独立性和逻辑独立性两个方面体现。从物理独立性角度上讲,用户的应用程序与存储在磁盘上的数据库是相互独立的,当数据的存储结构(或物理结构)改变时,通过对映像的相应改变可以保持数据的逻辑结构不变,从而应用程序也不必改变;从逻辑独立性角度上讲,用户的应用程序与数据库的逻辑结构是相互独立的,应用程序是依据数据的局部逻辑结构编写的,即使数据的逻辑结构改变了,应用程序也不必修改。

### 4. 数据由数据库管理系统统一管理和控制

数据库管理系统提供以下几个方面的数据控制功能。

#### (1) 数据库的安全性(Security)保护

保护数据以防止不合法的使用造成的数据泄密和破坏。

#### (2) 数据的完整性(Integrity)检查

数据的完整性是指数据的正确性和一致性,完整性检查是指将数据控制在有效的范围内,或保证数据之间满足一定的关系。

#### (3) 并发(Concurrency)控制

当多个用户的并发进程同时存取、修改数据库时,可能会发生相互干扰而得到错误的结果或使数据库的完整性和一致性遭到破坏,因此,必须对多用户的并发操作加以控制和协调。

#### (4) 数据库恢复(Recovery)

当计算机系统遭遇硬件故障、软件故障、操作员误操作或恶意破坏时,可能会导致数据错误或数据全部、部分丢失,此时要求数据库具有恢复功能。所谓的数据库恢复是指数据库管理系统将数据库从错误状态恢复到某一已知的正确状态,即完整性状态。

## 1.2 数据模型概述

现实世界中的数据要进入数据库中,需要经过人们的认识、理解、整理、规范和加工。可以把这一过程划分成三个主要阶段,即现实世界阶段、信息世界阶段和机器世界阶段。现实世界中的数据经过人们的认识和抽象,形成信息世界。在信息世界中用概念模型来描述数据及其联系,概念模型按用户的观点对数据和信息进行建模,独立于具体的机器和数据库管理系统(DBMS)。根据所使用的具体机器和DBMS,需要对概念模型做进一步转换,形成在具体机器环境下可以实现的数据模型。数据库是按照一定的数据模型组织存储在一起的数据集合。数据模型是对现实世界的模拟,它反映了现实世界中的客观事物以及这些客观事物之间的联系。

### 1.2.1 概念模型

信息世界是现实世界在人们头脑中的反映。人们对现实世界的客观事物及其联系进行充分的认识、理解和分析,将其抽象为某种信息结构,就得到了关于现实世界的概念级的模型,即概念模型。这样就将现实世界抽象为信息世界。在信息世界,用概念模型反映客观事物及事物间的联系。

### 1. 概念模型的名词术语

#### (1) 实体(Entity)

客观存在并可相互区别的事物称为实体。实体既可以是实际的事物,也可以是抽象的概念或联系。例如,学生、课程等就是实体。

#### (2) 属性(Attribute)

属性就是实体所具有的特性,一个实体可以用若干个属性描述。例如,用学号、姓名、性别、出生时间等描述学生实体,它们就是学生的属性,而课程的属性可以包括课程号、课程名、学分等。

#### (3) 域(Domain)

属性的取值范围称为该属性的域。例如,学生的性别只能取“男”或“女”。

#### (4) 实体型(Entity Type)

用实体名及其属性名集合抽象地刻画同类实体即为实体型。

#### (5) 实体集(Entity Set)

具有相同属性的实体的集合成为实体集。例如,全体学生即为实体集。

#### (6) 键(Key)

键能够唯一地标识一个实体集中每一个实体的属性或属性组合,也被称为关键字或码。例如,学生的学号,每一个学号都唯一地对应一个学生,没有两个学号相同的学生,也不会有在籍学生没有学号的情况。

#### (7) 联系(Relationship)

联系有两种,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。

### 2. 实体之间的联系

#### (1) 一对一联系

如果对于实体集 A 中的每个实体,实体集 B 中有且仅有一个(可以没有)实体与之相对应;相反的,对于实体集 B 中的一个实体,实体集 A 中同样有且仅有一个实体与之对应,则称实体集 A 与实体集 B 具有一对一联系,记作 1:1,如图 1.1(a)所示,例如,飞机票和乘客的关系即为一对一联系。

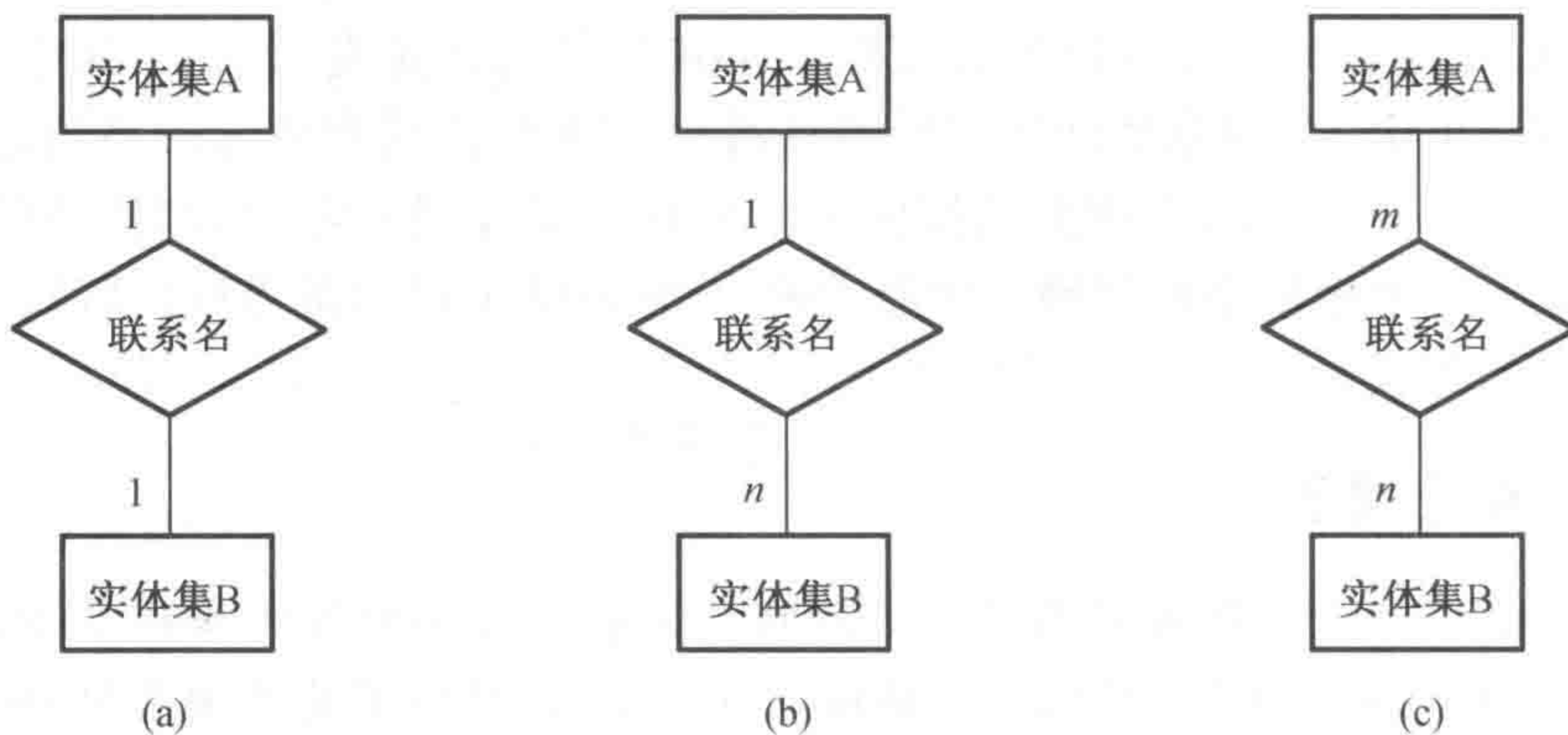


图 1.1 实体之间的三种联系

(a) 1:1 联系;(b) 1:n 联系;(c) m:n 联系

### (2) 一对多联系

如果对于实体集 A 中的每个实体,实体集 B 中有多个实体( $n \geq 0$ )与之相对应;反过来,对于实体集 B 中的每个实体,实体集 A 中至多只有一个实体与之相对应,则称实体集 A 与实体集 B 具有一对多联系,记作  $1:n$ ,如图 1.1(b)所示,例如,辅导员和班级的关系即为一对多联系。

### (3) 多对多联系

如果对于实体集 A 中的每个实体,实体集 B 中有多个实体( $n \geq 0$ )与之相对应;反过来,实体集 B 中的每个实体,实体集 A 中也有多个实体( $m \geq 0$ )与之相对应,则称实体集 A 与实体集 B 具有多对多联系,记作  $m:n$ ,如图 1.1(c)所示,例如,老师和学生的关系即为多对多联系。

## 3. E-R 模型

E-R 模型(Entity-Relationship)即实体-联系模型,是 1976 年由 P. P. S. Chen 提出的。这种模型用 E-R 图表示实体及其联系,广泛用于数据库设计中。E-R 图由实体、属性和联系三个基本要素组成。

### (1) 实体

实体即现实世界存在的、可以相互区别的人或事物。一个实体集合对应于数据库中的一个表,一个实体对应于表中的一行。实体用矩形表示,矩形内标注实体名称。

### (2) 属性

属性表示实体或联系的某种特征。一个属性对应于数据库表中的一列,也称为一个字段。用椭圆表示属性,椭圆内标注属性名称,并用连线与实体连接起来。

### (3) 联系

联系即实体之间的联系,在 E-R 图中用菱形表示,菱形内注明联系名称,并用连线分别将菱形框与相关实体相连,且在连线上注明联系类型,类型包括  $1:1$ ,  $1:n$  和  $m:n$  三种。

图 1.2 是用 E-R 图表示的学校教师授课情况的概念模型。

- ① 教师的属性有所属系、姓名等;
- ② 课程的属性有课程编号、名称等;
- ③ 学生的属性有学号、姓名、班级等。

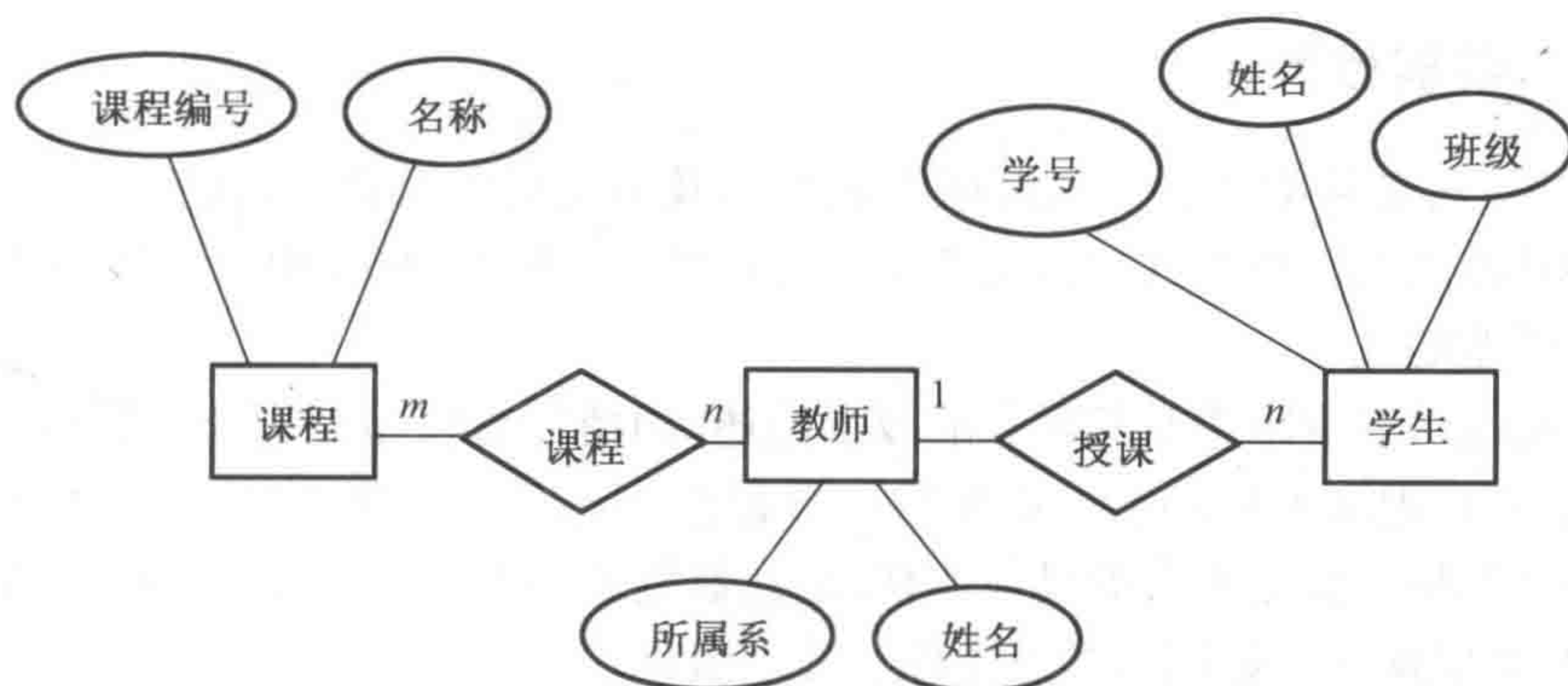


图 1.2 教师授课情况 E-R 图

E-R图直观易懂,是系统开发人员和客户之间很好的沟通媒介。对于客户(系统应用方)来讲,它概括了设计过程、设计方式和各种联系;对于开发人员来讲,它从概念上描述了一个应用系统数据库的信息组织。因此,如果能准确地画出应用系统的E-R图,就意味着彻底搞清了问题,以后就可以根据E-R图,结合具体的DBMS类型,把它演变为该DBMS所能支持的结构化数据模型。这种逐步推进的方法如今已经普遍应用于数据库设计中,E-R图成为数据库设计中的一个重要步骤。

**【例 1.1】** 为某企业设计一个E-R模型。

企业管辖若干连锁店铺,每家店铺经营若干商品,每家店铺有若干职工,但每个职工服务于一家店铺。

- ①店铺的的属性包括编号、店名、店址、店经理;
- ②商品的属性包括编号、商品名、单价、产地;
- ③职工的属性包括职工编号、职工姓名、性别、工资。

在联系中应反映出职工参加工作的时间、店铺销售商品的月销售量等。企业的店铺、商品及职工构成的E-R图如图1.3所示。

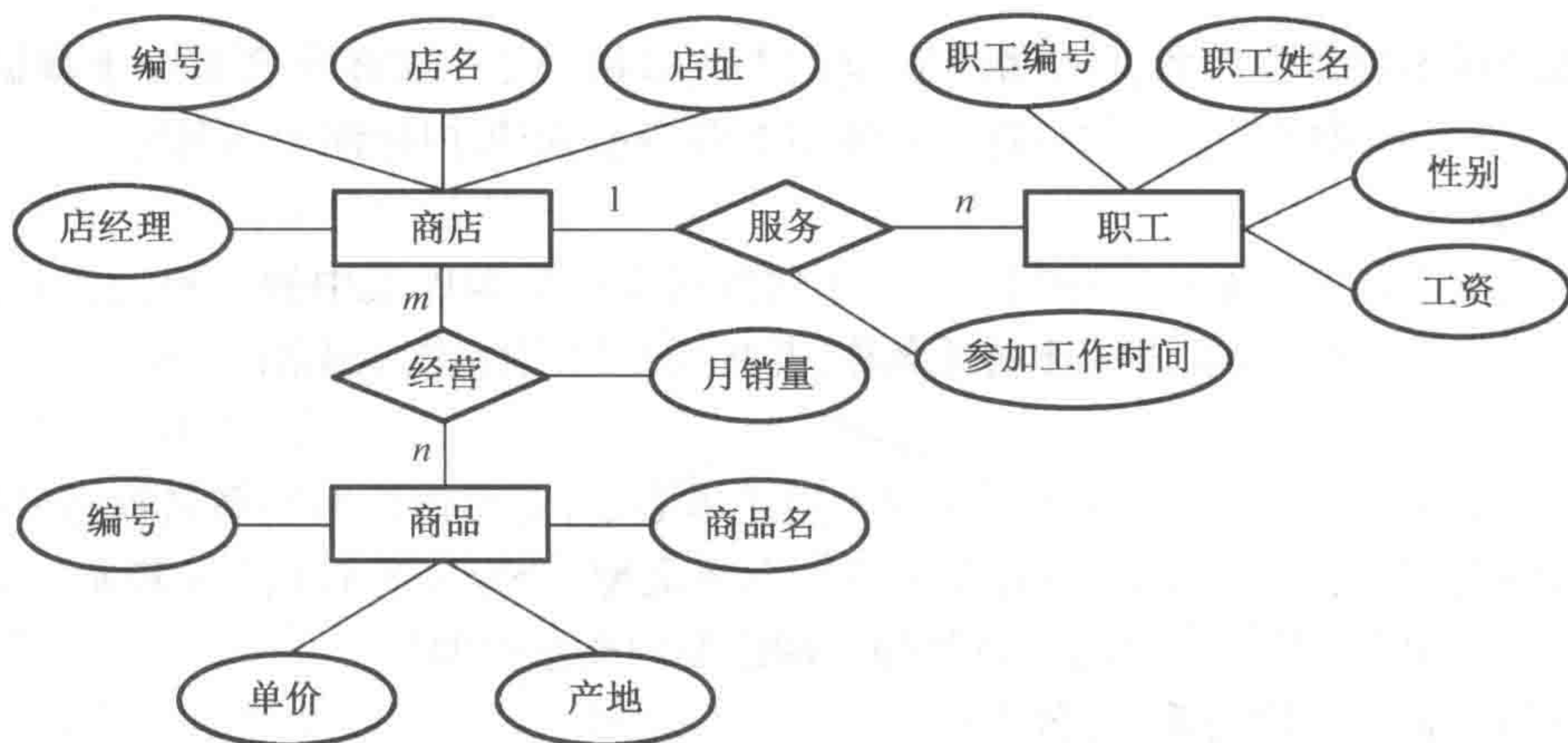


图 1.3 企业 E-R 图

### 1.2.2 数据模型

概念模型,只是从本质上直接反映客观事物及事物间的联系,并没有考虑在计算机上数据库中的具体实现。要将这种描述表示在计算机中,需要将概念模型转换为数据模型。

#### 1. 数据模型的概念

数据模型是对客观事物及其联系的数据描述,即使概念模型数据化。它描述数据以及数据间的联系,是现实世界的第二级抽象。数据模型不仅反映客观事物及事物间的联系,也考虑了在计算机上数据库中的具体实现,是在数据库中真正实现的模型。数据模型通常由数据结构、数据操作和数据的约束条件三部分组成。

##### (1) 数据结构

数据结构是所研究的对象类型的集合,这些对象组成了数据库,它们包括两类:一类是与数据类型、内容、性质有关的对象;另一类是与数据之间的联系有关的对象。按照数据结

构类型的不同,又可以将数据模型划分为层次模型、网状模型和关系模型。

## (2) 数据操作

数据操作是指对数据库中各种对象实例的操作。

## (3) 数据的约束条件

数据的约束条件是指完整性规则的集合。数据模型应反映和规定其必须遵守的、基本的、通用的完整性约束条件。数据的完整性约束是指在给定的数据模型中,数据及其数据关联所遵守的一组规则,用以保证数据库中数据的正确性和一致性。

## 2. 几种主要的数据模型

目前,应用于数据库技术中的数据模型有层次模型、网状模型、关系模型和面向对象数据模型。相对于不同的数据模型,数据库分为不同的类型:与层次模型相对应的数据库称为层次数据库;与网状模型相对应的数据库称为网状数据库;与关系模型相对应的数据库称为关系数据库;与面向对象数据模型相对应的数据库称为面向对象数据库。目前,市场上广为流行的是关系数据库以及以面向对象数据模型为基础的面向对象数据库。

### (1) 层次模型

在层次模型中,每个节点表示一个记录类型,记录类型之间的联系用节点之间的连线(有向边)表示,这种联系是父子之间的一对多的联系。层次数据库系统只能处理一对多的实体联系。

层次模型的一个基本特点是,任何一个给定的记录值只有按其路径查看时,才能显示出它的全部意义,没有一个子女记录值能够脱离双亲记录值而独立存在。如图 1.4 所示为层次模型示例。

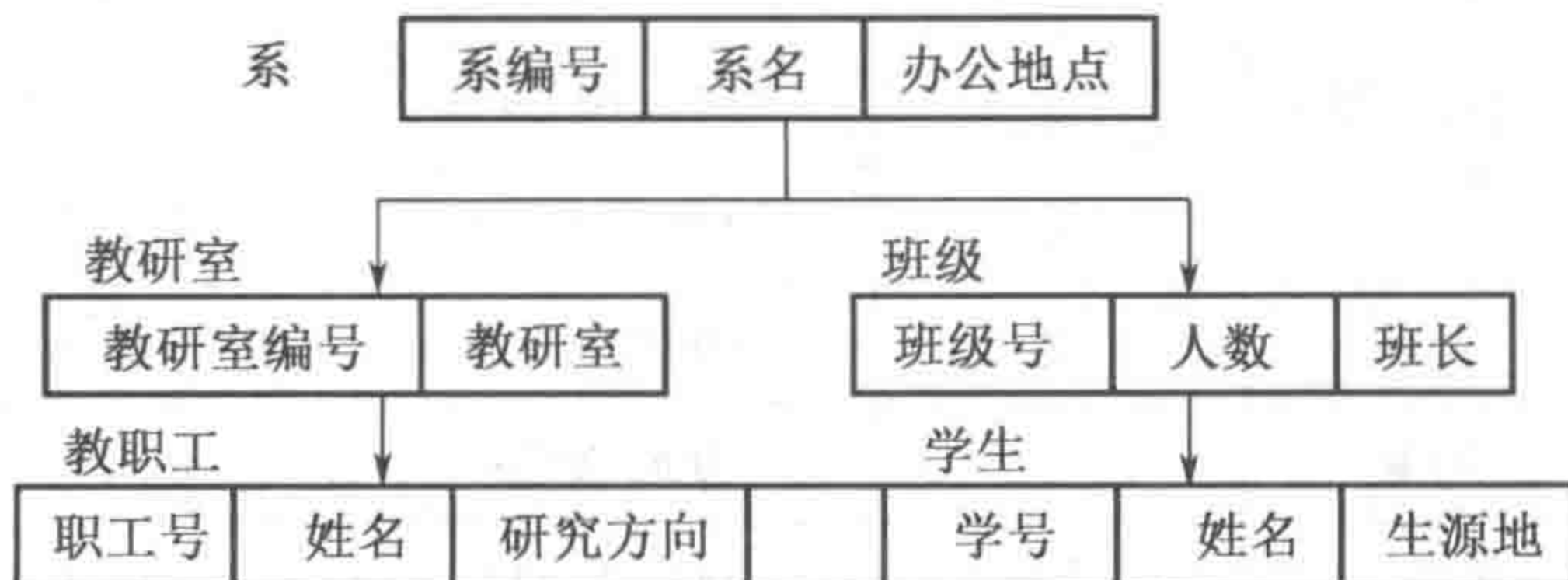


图 1.4 层次模型示例

层次模型反映实体间的一对多的联系。层次模型的优点是层次分明、结构清晰,适于描述客观事物中有主目、细目之分的结构关系;缺点是不能直接反映事物间多对多的联系,查询效率低。

### (2) 网状模型

现实世界中事物之间的联系更多的是非层次关系的,用层次模型表示这种关系很不直观。网状模型克服了这一弊病,可以清晰地表示这种非层次关系。网状模型消除了层次模型的两个限制,两个或两个以上的节点可以有多个父节点,此时有向树变成了有向图,该有向图描述了网状模型,如学生、课程、教室和教师间的关系。一个学生可以选修多门课程,一门课程可以由多个学生选修。如图 1.5 所示为网状模型示例。网状模型的优点是表达能力强,能更为直接地反映现实世界事物间多对多的联系,缺点是在概念上、结构上和使用上都比较复杂,数据独立性较差。

### (3) 关系模型

关系数据模型是由 IBM 公司的 E. F. Codd 于 1970 年首次提出的,以关系数据模型为基础的数据库管理系统,称为关系数据库管理系统(RDBMS),目前被广泛使用。关系模型是建立在数学概念上的,与层次模型、网状模型相比,关系模型是一种最重要的数据模型,主要由关系数据结构、关系操作集合、关系完整性约束三部分组成。实际上,关系模型可以理解为

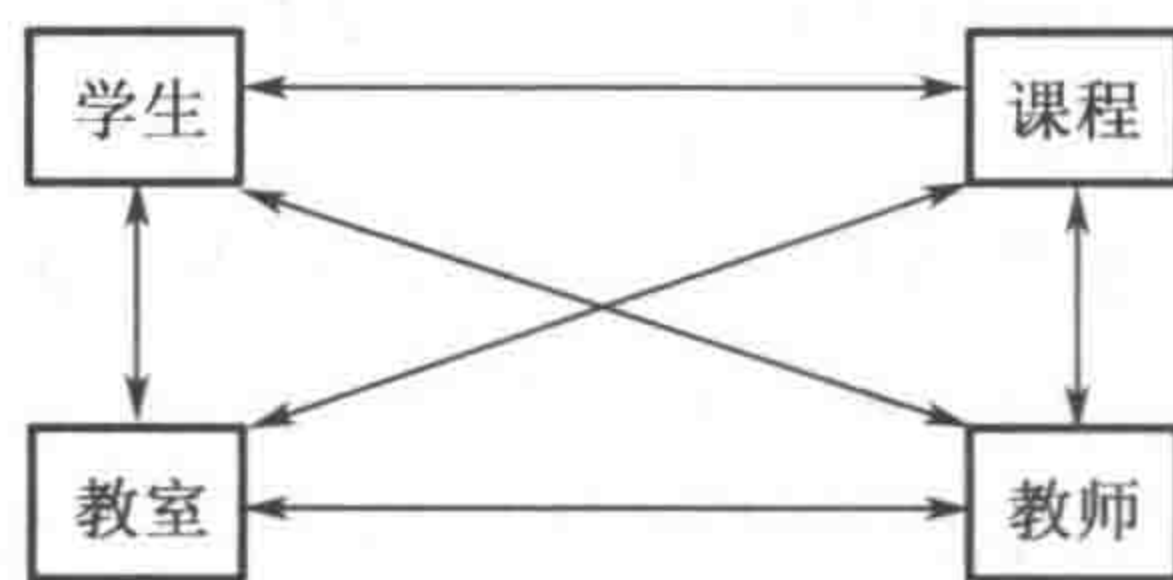


图 1.5 网状模型示例

用二维表格结构来表示实体及实体之间联系的模型,表格的列表示关系的属性,表格的行表示关系中的元组。在日常生活中,我们经常会碰到像花名册、工资单和成绩单等这样的二维表格,这些二维表的共同特点是由许多行和列组成,列有列名,行有行号。关系中的每一行称为一个元组。例如,表 1.1 中,“2015010101,秦家,男,1995/5/5,……”是一个元组。

表 1.1 学生情况表

学号	姓名	性别	出生时间	专业	总学分
2015010101	秦家	男	1995/5/5	电子信息工程	20
2015010102	李班	男	1995/12/12	电子信息工程	20
2015010103	刘忻	女	1994/12/27	电子信息工程	20
2015010105	马娇	女	1995/6/13	电子信息工程	18
2015010106	朱可	女	1995/7/1	电子信息工程	20
2015010107	赵翊辰	女	1994/9/21	电子信息工程	20
2015010108	王丹	男	1995/11/12	电子信息工程	16
2015010109	李楠	女	1995/1/12	电子信息工程	16
2015010110	刘丽	女	1996/2/23	电子信息工程	20
2015030101	马学文	男	1994/12/14	网络工程	19
2015030102	张明	男	1995/9/23	网络工程	19
2015030103	王丽男	男	1995/9/23	网络工程	12
2015030105	贾志	男	1994/10/22	网络工程	19
2015030106	徐元丹	女	1994/11/11	网络工程	19

关系中的每一列称为一个属性。例如,表 1.1 中,“学号”列是一个属性,“姓名”列也是一个属性。关系中能够唯一确定一个元组的属性或属性组合称为关键字。

例如,表 1.1 中每个学生的学号各不相同,学号可以唯一确定一个元组,因此,学号就是该关系的关键字,也就是主键。

对关系的描述一般为:关系名(属性 1,属性 2,……,属性 n),称为关系模式,例如学生(学号,姓名,性别,出生时间,专业)。关系模型的数据结构简单、概念清楚,符合人们的思维习惯,表达能力强,能直接反映实体间的三种联系,并且建立在严格的数学理论上。

因此,关系模型是目前使用最为广泛的一种数据模型,以关系模型为基础建立的关系数据库是当前市场上最为流行的数据库。

#### (4) 面向对象模型

面向对象数据模型是基于面向对象程序设计中所支持的对象语义定义的逻辑数据模型,它是持久的和共享的对象集合,具有模拟整个解决方案的能力。面向对象数据模型把实体表示为类,一个类描述了对象属性和实体行为。例如,CUSTOMER类,它不仅含有客户的属性(如客户编号、客户姓名和客户地址等),还包含模仿客户行为(如修改订单)的过程。类对象的实例对应于客户个体。在对象内部,类的属性用特殊值区分每个客户(对象),但所有对象都属于类,共享类的行为模式。面向对象数据库通过逻辑包含(Logical Containment)维护联系,面向对象数据库把数据和与对象相关的代码封装成单一组件,外面不能看到里面的内容。因此,面向对象数据模型强调对象(由数据和代码组成)而不是单独的数据,这主要是从面向对象程序设计语言继承而来的。在面向对象程序设计语言里,程序员可以定义包含其自身的内部结构、特征和行为的新类型或对象类。

## 1.3 关系数据库的基本原理

关系数据库是以关系模型为数据模型的数据库。关系模型建立在严格的数学理论基础之上,它将用户数据的逻辑结构归纳为满足一定条件的二维表的形式。关系数据库的建立,关键在于构造设计合适的关系模型。

### 1.3.1 关系模型

前面对关系模型做了直观的描述,关系模型是建立在数学概念上的,与层次模型、网状模型相比,关系模型是一种最重要的数据模型,它主要由关系数据结构、关系操作集合和关系完整性约束三部分组成。

#### 1. 关系模型的基本概念

一个关系对应于一张二维表,这个二维表是指包含有限个不重复行的二维表。在对E-R模型的抽象上,每个实体集和实体间的联系在这里都转化为关系或称二维表,而E-R模型中的属性在这里转化为二维表的列,也可称为属性,每个属性的名称称为属性名,也可以称为列名。每个属性的取值范围称为该属性的域。对二维表中每个属性取值后形成的一行数据称为该二维表的一个元组(行)。实际上,关系模型可以理解为用二维表格结构表示实体及实体之间联系的模型,表格的列表示关系的属性,表格的行表示关系中的元组。关系模型允许定义四类完整性约束:实体完整性、域完整性、参照完整性和用户定义的完整性。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;域完整性是关系中的列必须满足的某种特定的数据类型或约束;用户定义的完整性是应用领域需要遵循的约束条件。

#### 2. 关系的性质

关系是一种规范化了的二维表中行的集合。为了使相应的数据操作简化,在关系模型中对关系进行了限制,因此,关系具有以下六条性质:

- ①列是同质的,即每一列中的分量都是同一类型的数据,来自同一个域;
- ②关系中的任意两个元组不能相同;

- ③关系中不同的列,应该来自不同的域,每一列有不同的属性名;
- ④关系中各列的顺序可以任意互换,不会改变关系的意义;
- ⑤关系中各行的次序和列的次序一样,也可以任意交换;
- ⑥关系中每一个分量都必须是不可分的数据项,属性和元组分量具有原子性。

### 3. 关系的完整性

在阐述关系的完整性之前,首先介绍几个术语。

#### (1) 候选键(Candidate Key)

如果关系中的某一属性或属性组的值能唯一地标识一个元组,则称该属性或属性组为候选键。

#### (2) 主键(Primary Key)

若一个关系中有多个候选键,则选定一个为主键,主键也称为主码。

#### (3) 主属性(Primary Attribute)

主键的属性称为主属性。

#### (4) 外键(Foreign Key)

设 F 是基本关系 R 的一个属性或属性组合,但不是 R 的键(主键或候选键),如果 F 与基本关系 S 的主键 K 相对应,则称 F 是 R 的外键,并称 R 为参照关系,S 为被参照关系,外键也称为外码。关系的完整性有四类:实体完整性、参照完整性、域完整性和用户定义的完整性,下面分别介绍。

#### 1. 实体完整性

实体完整性规则是指若属性 A 是基本关系 R 的主属性,则属性 A 不能取空值,并且是唯一的。实体完整性规则规定基本关系的所有主属性项都不能取空值(NULL),而不仅是主属性整体不能取空值,并且具有唯一性。空值就是“不知道”或“无意义”。

**【例 1.2】** 有如下关系模式:

①学生(学号,姓名,性别,出生时间,专业,总学分,照片,备注),其中,学号属性为主码,不能取空值,并且所有学生的学号值必须各不相同。

②成绩(学号,课程号,成绩),其中,学号、课程号属性组合为主码,两者都不能取空值。

#### 2. 参照完整性

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描述的,这样就自然存在着关系与关系间的引用。先看下面的例子。

**【例 1.3】** 在学生管理关系数据库中,包括学生关系 xs,课程关系 kc 和成绩关系 cj 这三个关系分别为:

xs(学号,姓名,性别,出生日期,专业,总学分,照片,备注);

kc(课程号,课程名,学分,学时数);

cj(学号,课程号,成绩)。

这三个关系之间也存在着属性的引用,即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”。显然,成绩关系中的学号值必须是确实存在的学生的学号,即学生关系中有该学生的记录;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,即课程关系中有该课程的记录。换句话说,成绩关系中学号和课程号属性的取值,必须参照其他两个关系的相应属性的取值。参照完整性规则定义了一个关系数据库中不同表中列之间的关系,即外码与主码之间的引用规则。要求外键属性值只能取主表中的主键属性值



或空值,不能引用主键表中不存在的属性值,同时,如果一个主码值发生更改,则整个数据库中,对该值的所有引用要统一进行更改。

### 3. 域完整性

域完整性是指关系中的列必须满足某种特定的数据类型或约束。可以使用域完整性强制实现域完整性限制类型、限制格式或限制值的范围等。例如,限定性别列,只能取值“男”或“女”。

### 4. 用户定义完整性

用户定义的完整性,就是用户按照实际的数据库应用系统运行环境的要求,针对某一具体关系数据库定义的约束条件。例如,属性“成绩”的取值范围必须在0~100之间。用户定义完整性可以反映某一具体应用所涉及的数据必须满足的语义要求,保证数据库中数据取值的合理性。

## 1.3.2 关系运算

### 1. 传统的集合运算

传统的集合运算,包括并、交、差、广义笛卡儿积四种运算。设关系R和关系S具有相同的目 $n$ (即两个关系都具有 $n$ 个属性),且相应的属性取自同一个域,如图1.6所示,则四种运算定义如下。

A	B	C
a1	b1	c1
a1	b1	c1
a2	b2	c2

(a)

A	B	C
a1	b1	c1
a1	b1	c1
a2	b2	c2

(b)

图 1.6 关系 R 和关系 S

(a)关系 R;(b)关系 S

#### (1) 并

关系R与关系S的并由属于R或属于S的元组组成,其结果关系仍为 $n$ 目关系,记作 $R \cup S$ 。

#### (2) 交

关系R与关系S的交由既属于R又属于S的元组组成,其结果关系仍为 $n$ 目关系,记作 $R \cap S$ 。

#### (3) 差

关系R与关系S的差由属于R而不属于S的所有元组组成,其结果关系仍为 $n$ 目关系,记作 $R - S$ 。

#### (4) 广义笛卡儿积

两个分别为 $n$ 目和 $m$ 目的关系R和S的广义笛卡儿积是一个 $(n+m)$ 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 $a_1$ 个元组,S有 $a_2$ 个元组,则关系R和关系S的广义笛卡儿积有 $a_1 \times a_2$ 个元组,记作 $R \times S$ 。

**【例 1.4】** 已知关系R和关系S,如图1.6所示,求 $R \cup S, R \cap S, R - S, R \times S$ ,结果如图1.7和图1.8所示。