

# 第 1 单元

## 基 础 篇

### 【单元简介】

本单元是后续单元的基础，共 5 章，各章简介如下。

第 1 章：认识数据库，主要任务：了解并认识数据库体系结构、数据库关系模型及关系完整性约束。通过本章的学习，读者能识别数据库体系结构及构成要素、数据库管理系统的组成及功能、概念模型和数据模型特征与联系，能定义实体完整性和参照完整性。

第 2 章：数据库设计，主要任务：全面认识数据库设计、调研分析数据库需求、数据库概念结构设计、数据库逻辑结构设计、数据库物理结构设计，以及部署与维护数据库。通过本章的学习，读者能规划数据库设计流程与步骤，能对项目进行需求分析，能根据需求分析文档进行数据库的概念结构设计、逻辑结构设计、物理结构设计，能配合开展数据库的测试、运行和维护工作。

第 3 章：安装与配置 MySQL 数据库，主要任务：安装、配置及登录与连接 MySQL 数据库。通过本章的学习，读者能下载并安装 MySQL 数据库，能配置、启动与停止 MySQL 数据库系统，能登录与连接 MySQL 数据库，能熟练使用 1~2 种图形管理工具并在图形界面中操作 MySQL 数据库。

第 4 章：操作 MySQL 数据库与表，主要任务：增、删、改、查数据库和表。能创建、查看、删除、修改数据库，能创建、查看、删除、修改数据表，能对满足 WHERE 条件的记录进行插入、删除、修改，能创建及维护表的完整性约束。

第 5 章：查询 MySQL 数据，主要任务：单表查询、多表查询，以及 SELECT、WHERE、ORDER BY、GROUP BY、LIMIT 等查询子句的嵌套使用。通过本章的学习，读者能正确运用子查询语法规则描述条件查询表达式，能对单表或多表进行数据查询，能对查询结果进行合并输出。

# 第1章

## 认识数据库

### 1.1 情景引入

小李是某高校大一学生，身为班里的学习委员，他经常需要管理和统计本班学生的各类信息，这对小李来说不是一项轻松的工作。于是，他想能不能用计算机完成这些大量数据的常规统计和管理呢？听说数据库可以帮助管理大量数据，小李非常兴奋，希望能早日开发一个学生信息管理系统来解决每天让自己因采集各类数据而头疼的问题。为了完成这项开发任务，小李现已迫不及待地想要学习和了解数据库的相关知识和技术。

数据库技术是一种数据管理技术，产生于 20 世纪 60 年代，经过多年的发展，已经形成了自己的理论体系，成为计算机科学的一个重要分支。数据库技术解决了计算机信息处理过程中如何有效组织和存储海量数据的问题，体现了先进的数据管理思想，使计算机应用渗透到社会各个领域，在当今的信息社会中发挥着越来越大的作用。

数据库与大数据是什么关系呢？大数据是一门研究数据的科学，重点研究数据的规律和现象，包括数据的获取、存储、管理、分析及数据的可视化等；数据库是数据的一种存储方式，主要用于组织、存储数据。数据库种类较多，可分为 SQL 关系型数据库和 NoSQL 非关系型数据库。

本章将带读者一起了解数据库体系结构及相关知识。

### 1.2 任务目标

#### ➡ 知识目标

1. 了解信息与数据、数据库、数据库系统、数据库管理系统等术语的内涵及功能。
2. 掌握数据库管理系统的组成及功能。
3. 掌握数据模型的分类及特征。
4. 掌握数据关系完整性约束的分类及作用。

#### ➡ 能力目标

1. 能全面识别数据库体系结构的构成要素。
2. 能充分识别数据库管理系统的组成及功能。
3. 能充分识别概念模型和数据模型的特征与联系。

4. 能充分识别实体完整性和参照完整性。

### ➔ 素质目标

1. 具备一定的数据科学素养。
2. 具备一定的数据安全、规范及道德规范意识。
3. 具备一定的数据全局设计、规划及统筹能力。
4. 具备一定的数据设计创新能力和实践能力。
5. 具备较强的团队合作意识和服务精神。
6. 具备较强的自我管理能力和自学能力。
7. 具备较强的责任意识与担当精神。

## 1.3 任务实施

### 任务 1.3.1 认识数据库体系结构



微课视频

#### 1. 认识信息与数据

(1) 信息：信息是客观事物在人脑中的反映，是以各种方式传播的关于某一事物的消息、情报或知识等，泛指人类传播的一切内容。人通过获得、识别自然界和社会的不同信息来区别不同事物，得以认识和改造世界。随着现代科学技术的发展，生产力水平大大提高，在经济、文化、军事等领域里需要人们掌握大量的信息，研究和分析这些信息，从中得出有用结论，并将其应用到社会实践活动中。计算机的问世和发展给人们提供了用计算机处理和管理信息的可能。人们在使用计算机处理信息的同时开发了信息资源。信息同能量、物质并列为当今世界三大资源，是国民经济和社会发展的重要战略资源。信息资源的开发和利用会进一步促进社会及生产的发展。

(2) 数据：数据是描述客观事物的符号记录，是信息的表现形式和载体。在计算机系统中，各种字母、数字符号的组合与语音、图形、图像等统称为数据，数据经过加工就成为信息。在日常生活中，人们通常直接用自然语言描述事物信息，而在计算机中，为了存储和处理这些抽象的事物信息，人们通常抽取对这些事物感兴趣的特征值，并使用特定的符号加以描述。例如，在描述职工人事档案时，人们感兴趣的可能是职工的员工编号、姓名、性别、年龄、生日、籍贯、家庭住址、政治面貌、职称、行政职务等基本信息，针对这些信息可描述为：001、张三、女、36、1985、重庆市、重庆市渝中区、中共党员、高级工程师、处长。这里的职工人事档案记录就是数据。

(3) 数据语义：数据的含义称为数据的语义，例如，以上记录中的每个数据项必须经过解释才能明确其含义。上述记录可以解释为“姓名为张三的女性员工，1985 年出生，现年 36 岁”等。

数据与信息是不可分的，数据是信息的符号表示；信息则是数据的内涵，是对数据的语义解释。

(4) 数据管理：对信息数据进行收集、整理、组织、存储、传播、检索、分类、加工、计算、打印报表、输出等一系列活动可被称为数据信息处理或管理。数据管理是数据处理

的基本环节, 数据管理技术的优劣直接影响着数据处理的效果。数据库技术就是一种先进的数据管理技术。

## 2. 认识数据库

数据库(Database, DB)简单、形象地说, 就是电子化的文件柜, 是长期存储在计算机内的、有组织的、可共享的相关数据集合。数据库中保存的是以一定的组织方式存储在一起的具有相互关联的数据整体。也就是说, 数据库不仅保存数据, 还保存数据与数据之间的联系。数据库中的数据可以被多个应用程序的用户使用, 从而达到数据共享的目的。

数据库中的数据与应用程序之间可以彼此独立。在数据库应用系统中, 数据的组织和存储方法与应用程序互不依赖、相互独立。应用程序不再与一个孤立的数据文件相对应, 它所涉及的数据取自整体数据集合的某个子集, 作为逻辑文件与应用程序相对应, 并通过系统管理软件实现逻辑文件与物理数据之间的映射。

数据库中的数据是相互关联的。数据库中的数据不是孤立的, 数据与数据之间存在相互联系。在数据库中不仅存放了数据本身, 还存放了数据与数据之间的联系。例如, 在教学管理系统中, 数据库不仅存放了学生和课程的数据, 还存放了哪些学生选修了哪几门课程, 这就反映了学生数据与课程数据之间的联系。

综上所述, 数据库是以一定的组织方式存放在一起的, 能够被多个用户共享的, 与应用程序彼此独立、相互关联的数据集合。

需要明确的是, 数据库和数据仓库(Data Warehouse)不是同一个概念。数据仓库是在数据库技术的基础上发展起来的一个新的应用领域。

## 3. 认识数据库系统

### 1) 数据库系统的定义

数据库系统(Database Systems)是为了适应数据处理的需要而发展起来的一种较为理想的数据处理系统, 能够实现组织、存储大量相关数据。

### 2) 数据库系统构成

数据库系统一般包括数据、硬件、软件 and 用户 4 个部分。

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: 硬件是构成计算机系统的各种物理设备, 包括存储所需的外部设备。硬件的配置应满足整个数据库系统的需要。

③ 软件: 包括操作系统、数据库管理系统及应用程序。数据库管理系统是数据库系统的核心软件, 是在操作系统的支持下, 解决如何科学地组织和存储数据, 如何高效地获取和维护数据的系统软件。

④ 用户: 包括专业用户、非专业用户和数据库管理员。

➤ 专业用户指程序员, 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, 为最终用户开发应用程序。

➤ 非专业用户即最终用户, 是非计算机专业人员。他们通过应用系统提供的用户接口界面以交互方式操作使用数据库。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(Database Administrator, DBA), 是负责管理和维护数据库管理系统的人。数据库管理员负责全面管理和控制数据库系统。对于大型数据库系统, 则要



微课视频

求配备专门的数据库管理员。  
数据库系统结构如图 1.1 所示。

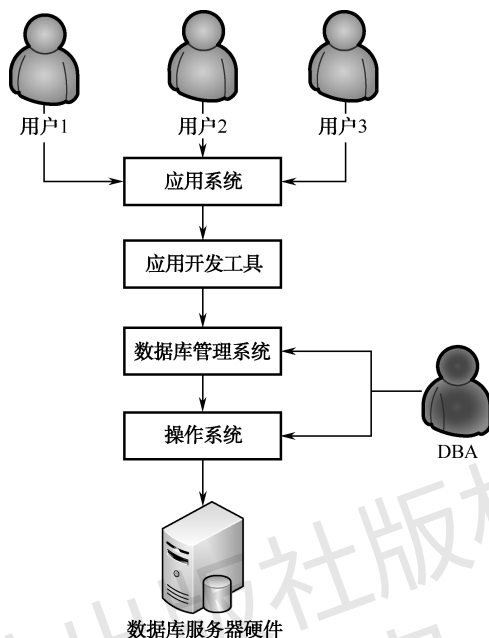


图 1.1 数据库系统结构



微课视频

#### 4. 认识数据库管理系统

数据库管理系统（Database Management System, DBMS）是一种操纵和管理数据库的大型软件，负责对数据库资源进行统一的管理和控制，其职能是建立数据库、维护数据库、接受并完成用户提出的访问数据的各种请求，并为数据库的安全性和完整性提供保证。DBMS 常见品牌包括：Sybase、DB2、Oracle、MySQL、Access、SQL Server、Informix、PostgreSQL 等。

DBMS 位于用户与操作系统之间，可以使多个应用程序和用户用不同的方法在同一时刻或不同时刻建立、修改和询问数据库。用户通过 DBMS 访问数据库中的数据，不必关注数据存放的细节，从而方便建立、使用和维护数据库；数据库管理员通过 DBMS 进行数据库的维护工作。DBMS 的主要组成及功能如下。

##### 1) 数据定义

DBMS 提供数据定义语言（Data Definition Language, DDL），主要用于建立、修改数据库的库结构，定义数据库的完整性约束条件和保证完整性的触发机制等。DDL 所描述的库结构仅仅给出了数据库的框架，数据库的框架信息被存放在数据字典（Data Dictionary）中。

##### 2) 数据操纵

DBMS 提供数据操作语言（Data Manipulation Language, DML），用户可以使用 DML 操纵数据，实现对数据库中数据的查询、插入、修改、删除等基本操作。国际标准数据库操作语言——SQL，就是 DML 的一种。

##### 3) 数据库运行管理与控制

DBMS 提供一系列系统运行控制程序，负责在数据库运行过程中对数据库进行管理和控制。所有访问数据库的操作都要在这些控制程序的统一管理下进行，以保证数据库系统的正常运行。



微课视频

#### 4) 数据组织、存储与管理

DBMS 要分类组织、存储和管理各种数据,包括数据字典、用户数据、存取路径等,需要确定以何种文件结构和存取方式组织这些数据,如何实现数据之间的联系。数据组织和存储的基本目标是提高存储空间利用率,并选择合适的存取方法提高存取效率。

#### 5) 数据库的保护

数据库中的数据是信息社会的战略资源,所以数据的保护至关重要。DBMS 对数据库的保护通过数据库的安全性控制、完整性控制、并发控制,以及数据库的恢复来实现。DBMS 还有系统缓冲区的管理及数据存储的某些自适应调节机制等其他保护功能。

#### 6) 数据库的维护

DBMS 对数据库的维护包括数据的载入、转换、转储,数据库的重组、重构及性能监控等功能,这些功能分别由各个应用程序来完成。

#### 7) 通信

DBMS 具有与操作系统的联机处理、分时系统及远程作业输入的相关接口,负责处理数据的传送。在网络环境下的数据库系统,还应该包括 DBMS 与网络中其他软件系统的通信功能和数据库之间的相互操作功能。

### 任务 1.3.2 认识数据库的模型

提到模型,人们自然会联想到建筑模型、飞机模型等事物。从广义来说,模型是现实世界特征的模拟和抽象。在数据库中,使用数据模型(Data Model)对现实世界进行抽象。数据模型应满足以下 3 方面要求:一是能比较真实地模拟现实世界;二是容易被人理解;三是便于在计算机上实现。

#### 1. 认识数据模型

在数据库系统中,针对不同的使用对象和应用目的,采用不同的数据模型。其中,根据模型的应用目的,可以将数据模型分为概念数据模型、逻辑数据模型和物理数据模型。

(1) 概念数据模型:简称概念模型,主要用来描述世界的概念化结构,使数据库的设计人员在设计的初始阶段,摆脱计算机系统及 DBMS 的具体技术问题,集中精力分析数据及数据之间的联系等,与具体的 DBMS 无关。概念数据模型只有转换成逻辑数据模型,才能在 DBMS 中实现。

(2) 逻辑数据模型:简称数据模型,是用户从数据库中看到的模型,是具体的 DBMS 所支持的数据模型,如网状数据模型(Network Data Model)、层次数据模型(Hierarchical Data Model)等。此模型既要面向用户,又要面向系统,主要用于 DBMS 的实现。

(3) 物理数据模型:简称物理模型,是面向计算机物理表示的模型,描述了数据在储存介质上的组织结构,它不但与具体的 DBMS 有关,还与操作系统和硬件有关。每一种逻辑数据模型在实现时都有对应的物理数据模型。

下面重点介绍概念数据模型和逻辑数据模型。

#### 2. 认识概念数据模型

##### 1) 概念数据模型的相关术语

(1) 实体(Entity):客观存在并可相互区别的事物被称为实体。实体可以是人,可以



微课视频

是物，也可以是事；可以是实际对象，也可以是概念；可以是事物本身，可以是事物之间的联系，如一名学生、一辆轿车、一张椅子、一个部门等，也可以是抽象的事件，如一次足球比赛、一次借书等。

(2) 属性 (Attribute)：实体所具有每个特性被称为属性。例如，学生实体可以由“学号、姓名、专业名、性别、出生日期、身高”等属性组成。比如，“101101，林琳，计算机软件，男，1991-8-10，175.5cm”这些属性组合起来表征了一名学生。属性可以分为简单属性、复合属性、单值属性和多值属性。

- 简单属性 (Simple Attribute)：仅由单个元素组成的属性。简单属性是不能被进一步分解的。例如，实体“学生”的学号、身份证号就是简单属性。
- 复合属性 (Composite Attribute)：由多个元素组成的属性，复合属性可以被进一步分解为多个独立存在的更小元素。例如，实体“学生”的姓名可以分为曾用名和现用名。
- 单值属性 (Single-Valued Attribute)：实体的某个属性只有一个值。对具体的实体来说，大多数实体是单值属性。例如，实体“学生”的身份证号就只有一个，如 411213×××3214。
- 多值属性 (Multi-Valued Attribute)：实体的某个属性可以有多个值。例如，学生的个人爱好就可能有多值，如篮球、足球等。

(3) 关键字 (Key)：能唯一地标识一个实体的属性的集合被称为关键字 (或码)。例如，学生的学号就是实体“学生”的关键字。

(4) 域 (Domain)：每个属性都有一个取值范围，被称为该属性的值域。值域的类型可以是整型、实型或字符型等。例如，年龄的值域为整数，性别的值域为 (男，女)，学号的值域为若干个数字构成的字符串集合，姓名的值域为字符串集合。

(5) 实体型 (Entity Type)：一类实体所具有的共同特征或属性的集合被称为实体型。一般用实体名及其属性来抽象地刻画一类实体的实体型。例如，学生 (学号、姓名、专业名、性别、出生日期、身高) 就是一个实体型。

(6) 实体集 (Entity Set)：同类型实体的集合被称为实体集。例如，全体学生、所有汽车、所有学校、所有课程、所有零件都被称为实体集。由此可知：事物若干属性值的集合可表征一个实体，而若干个属性所组成的集合可表征一个实体的类型，同类型实体的集合组成实体集。

(7) 联系 (Relationship)：现实世界的事物普遍存在两类联系，一类是实体内各属性之间的联系；另一类是各种实体之间的联系。在考虑实体内部联系时，可以把属性看作实体。两个实体之间的联系一般可分为以下 3 种。

- 一对一 (1:1) 联系：若对于实体集 A 中的每一个实体，实体集 B 中都有唯一的一个实体与之联系，反之亦然，则称实体集 A 与实体集 B 具有一对一联系，记作 1:1，如图 1.2 所示。

例如，在学校一个班只有一个正班长，而一个班长只在一个班级中任职，则班级与班长具有一对一联系。观众与座位、乘客与车票、病人与病床、学校与校长都具有有一对一联系。

- 一对多 (1:n) 联系：若对于实体集 A 中的某个实体，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 中只有一个实体与之联系，则称实体集 A 与实体集 B 具有一对多联系，记作 1:n，如图 1.3 所示。

例如，一个班级中有若干名学生，而一名学生只能在一个班级中学习，则班级与学生具有一对多联系；宿舍与学生也具有一对多联系。

- 多对多 ( $m:n$ ) 联系：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 1.4 所示。

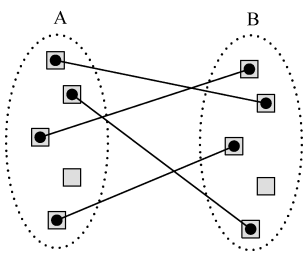


图 1.2 一对一联系

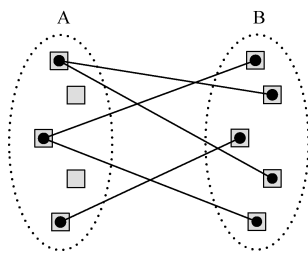


图 1.3 一对多联系

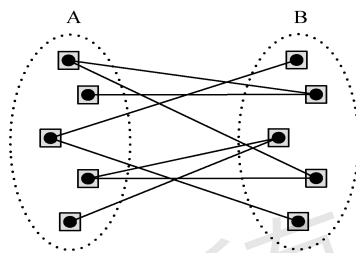


图 1.4 多对多联系

例如，一名学生可以选修若干门课程，而一门课程也可以有若干名学生选修，所以学生与课程具有多对多联系。

## 2) 概念数据模型的表示方法

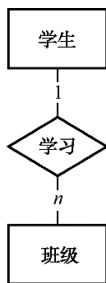
最常用的是实体-联系方法 (Entity-Relationship Approach)，简称 E-R 方法。该方法是由 P.P.S.Chen 在 1976 年提出的。E-R 方法采用 E-R 图来描述某一组织的概念数据模型，在这里仅介绍 E-R 图的要点。

- 长方形框：表示实体集，框内写上实体名称。
- 椭圆形框：表示实体属性，并用无向边把实体框及其属性框连接起来。
- 菱形框：表示实体间的联系，框内写上联系名，用无向边把菱形框及其有关的实体框连接起来，并标明联系的种类 ( $1:1$ ,  $1:n$  或  $m:n$ )。如果联系也具有属性，则把属性框和菱形框也用无向边连接上。

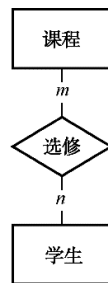
例如，学生、班长、班级、课程等实体间的 E-R 图，如图 1.5 所示。



(a) 一对一模型



(b) 一对多模型



(c) 多对多模型

图 1.5 实体间的 E-R 图

## 3) E-R 图的实例讲解

**【例 1.1】** 使用 E-R 图表示工厂库房管理系统的概念数据模型。

该库房信息如下。

- 库房：编号、名称、容量。
- 零件：零件号、零件名、型号规格。



➤ 职工：职工号、职工名、工种。

其中，每个库房有若干个职工，每个职工只能在一个库房工作；每个库房可存放若干种零件，每种零件可存放在不同的库房中。

因此，为系统建立概念数据模型其实就是为其设计对应的 E-R 图，其关键步骤如下。

第一步，确定实体和实体属性的 E-R 图，如图 1.6 所示。

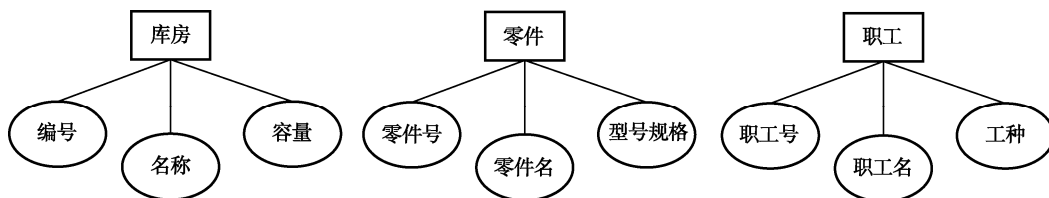


图 1.6 实体和实体属性的 E-R 图

第二步，确定实体之间的联系及联系类型的 E-R 图，如图 1.7 所示。

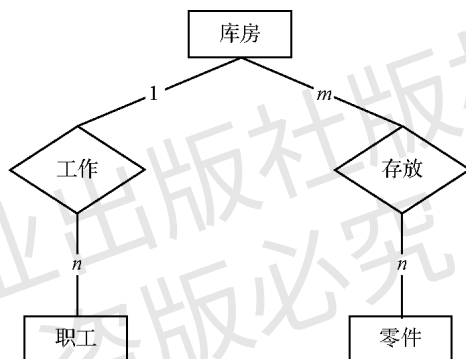


图 1.7 实体之间的联系及联系类型的 E-R 图

第三步，给实体和联系加上属性，即可完成工厂库房管理系统的 E-R 图，如图 1.8 所示。

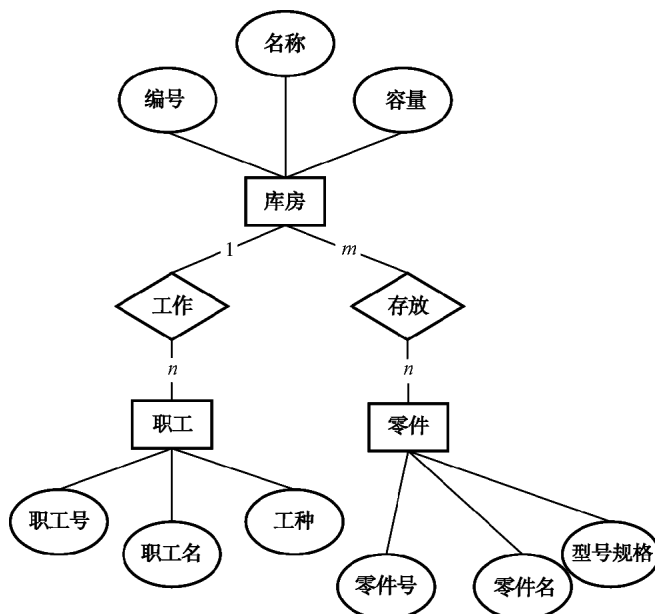


图 1.8 工厂库房管理系统的 E-R 图

**【例 1.2】** 假设某公司在多个地区设有分公司，经销本公司的各种产品，每个分公司聘用多个职工，且每个职工只属于一个分公司。分公司有公司名称、地区和联系电话等属性，产品有编码、名称和价格等属性，职工有编号、姓名和性别等属性，每个分公司销售产品有数量属性。要求根据上述语义画出 E-R 图。

根据题意可以确定：实体为分公司、职工和产品；分公司与职工之间为一对多（1： $n$ ）的联系，分公司与产品之间是多对多（ $m$ ： $n$ ）的联系；数量应该作为分公司与产品之间联系的属性。则该公司销售系统的 E-R 图，如图 1.9 所示。

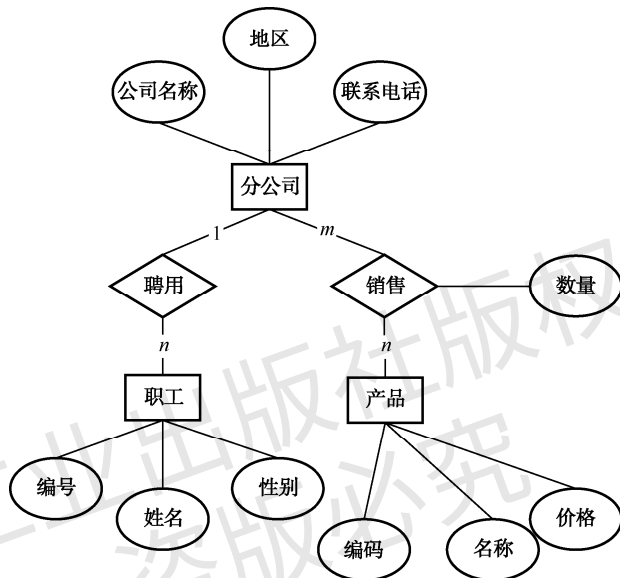


图 1.9 该公司销售系统的 E-R 图

### 3. 认识逻辑数据模型

逻辑数据模型反映的是系统分析设计人员对数据存储的观点。逻辑数据模型对概念数据模型进行了进一步的分解和细化，是根据业务规则确定关于业务对象、业务对象的数据项及业务对象之间关系的基本蓝图。逻辑数据模型的内容：所有的实体和关系，确定每个实体的属性，定义每个实体的主键，指定实体的外键，范式化处理等。逻辑数据模型的目标：尽可能详细地描述数据，但不考虑数据在物理上如何实现。逻辑数据建模不仅会影响数据库设计的方向，还会间接影响最终数据库的性能和管理。如果在实现逻辑数据模型时投入得足够多，那么在物理数据模型设计时就会有許多可供选择的方法。

逻辑数据模型是由数据结构、数据操作和数据约束 3 部分组成的。

- 数据结构：数据结构主要描述数据的类型、内容、性质，以及数据之间的联系，是整个逻辑数据模型的基础。而针对数据的操作和数据之间的约束都是建立在数据结构基础上的。
- 数据操作：主要定义了相应的数据结构上的操作类型和操作方式。比如，数据库中的增、删、改、查等操作。
- 数据约束：数据约束主要用来描述数据库中数据结构之间的语法、词义联系，以及彼此之间的相互约束和制约关系。比如，在 MySQL 数据库中使用外键保证数据之间的数据完整性。

目前, 数据库领域采用的逻辑数据模型有层次模型、网状模型和关系模型, 其中应用最广泛的是关系模型。

### 1) 层次模型

层次模型是最早用于商品数据库管理系统的逻辑数据模型, 采用树形结构表示实体之间的联系。其典型代表是于 1968 年问世的, 由 IBM 公司开发的数据库管理系统——IMS 层次数据库。在现实世界中, 许多实体集之间的联系就是一种自然的层次关系。例如, 行政机构、家族关系等都是层次关系。图 1.10 展示的就是学校中的部门层次模型。

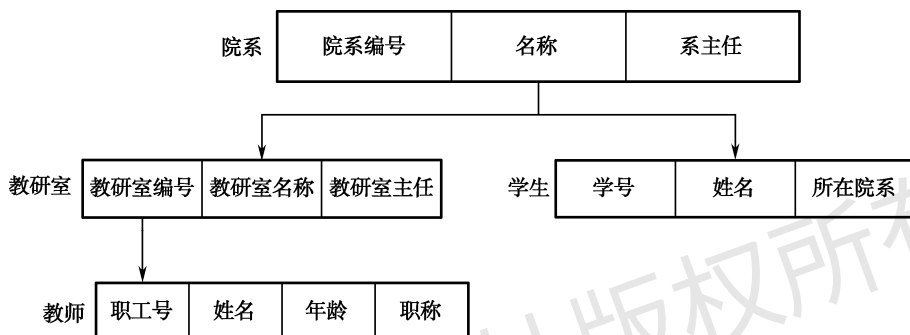


图 1.10 层次模型示例

层次模型的表示方法: 树的节点表示实体集(记录的实体型), 节点之间的连线表示相连两实体集之间的关系, 这种关系只能是“1:n”的。通常把表示“1”的实体集放在上方, 称为父节点; 表示“n”的实体集放在下方, 称为子节点。层次模型具有以下特征。

- 有且仅有一个根节点。
- 除根节点以外的其他节点有且仅有一个父节点。

因此, 层次模型只能表示“1:n”的关系, 而不能直接表示“m:n”的关系。

在层次模型中, 一个节点可以称为一个记录类型, 用来描述实体集。每个记录类型可以有一个或多个记录值, 即上层的一个记录值能对应下层的一个或多个记录值, 而下层的每个记录值只能对应上层的一个记录值。

### 2) 网状模型

网状模型的典型代表是数据库任务组(Database Task Group, DBTG)。DBTG 是美国数据系统语言协会(Conference on Data System Languages, CODASYL)的下属机构, 它于 1971 年提出 DBTG 报告。DBTG 报告是一个网状模型的数据描述语言和数据操纵语言的规范文本。

网状模型是一个不加任何条件限制的采用有向图结构表示实体类型及实体联系的数据模型, 更是一种具有普遍性的结构。网状模型具有以下特征。

- 可以有任意一个节点无双亲。
- 至少有一个节点有一个以上的双亲。
- 允许两个节点之间有一种或两种以上的联系。

在网状模型的 DBTG 标准中, 基本结构为简单二级树, 这被称为系。系的基本数据单位是记录, 它相当于 E-R 图中的实体集, 记录又由若干数据项组成, 它相当于 E-R 图中的属性。图 1.11 所示为教师授课数据库的网状模型。

例如, 一名学生可以选修若干门课程, 某一课程可以被多名学生选修, 因此, 学生与

课程之间是多对多联系。DBTG 模型中不能表示记录之间的多对多联系,为此引进一名学生选课的连接记录,它由 3 个数据项组成,即学号、课程号、成绩,表示某名学生选修的某一门课程及其成绩。其网状模型如图 1.12 所示。

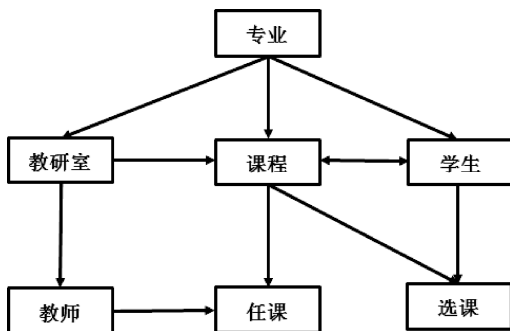


图 1.11 网状模型示例 (1)

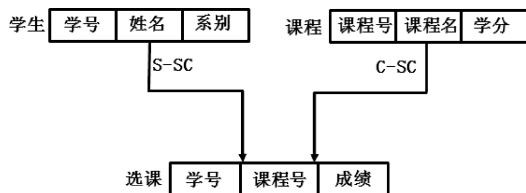


图 1.12 网状模型示例 (2)

综上所述,网状模型明显优于层次模型。网状模型在一定程度上支持数据的重构,具有一定的数据独立性和共享特性,且运行效率较高。但网状模型在应用时存在以下问题。

- 网状结构较为复杂,增加了用户查询和定位的难度。
- 网状数据操作命令具有过程式性质。
- 不直接支持对于层次结构的表达。

### 3) 关系模型

网状数据库和层次数据库已经很好地解决了数据的集中和共享问题,但是在数据独立性和抽象级别上仍有较大欠缺。用户在对这两种数据库进行存取时,仍需要明确数据的存储结构,指出存取路径。而后来出现的关系数据库较好地解决了这些问题。关系数据库理论出现于 20 世纪 60 年代末到 70 年代初。1970 年,IBM 的研究员 E.F.Codd 博士在发表的《大型共享数据银行的关系模型》一文中正式提出了关系模型的概念。

目前,关系模型是数据库领域中最重要的一种数据模型。关系模型的本质是一个二维表。在关系模型中,一个二维表就可以被称为一个关系,如表 1.1 就是一个关系。自 20 世纪 80 年代以来,计算机厂商推出的 DBMS 几乎都是关系型的,如 Oracle、Sybase、Informix、SQL Server、Visual FoxPro 等。

表 1.1 某班学生成绩表

| 学 生 学 号  | 姓 名 | 语文/分 | 数学/分 | 英语/分 |
|----------|-----|------|------|------|
| 20191001 | 张晶晶 | 89   | 67   | 90   |
| 20191002 | 王思雨 | 67   | 89   | 67   |
| 20191003 | 潘明明 | 90   | 90   | 87   |
| 20191004 | 秦小曼 | 76   | 60   | 89   |

注:虚构数据,仅用作图表示例。

下面以学生选课系统为例对关系模型进行说明。

**【例 1.3】** 学生选课系统的实体包括学生、教师、课程,其联系一般为:学生与课程之间是多对多联系,教师与课程之间也是多对多联系。学生可以同时选择多门课程,一门课程也可以同时被多名学生选择;一名教师可以教授多门课程,一门课程也可以由多名教师教授。因此,学生、教师、课程之间的联系如图 1.13 所示。

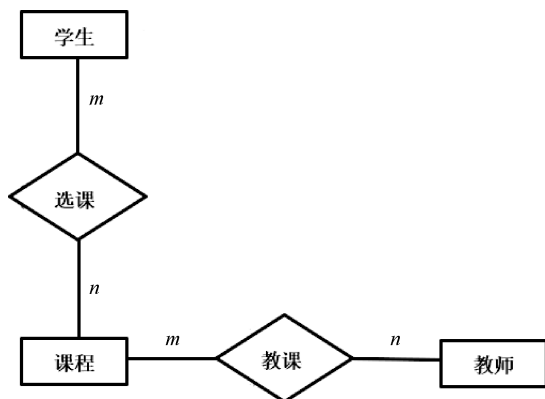


图 1.13 学生、教师、课程之间的联系

将图 1.13 映射为关系模型中的表格，如图 1.14 所示。从图 1.14 中可以看到学生与课程之间的联系，教师与课程之间的多对多联系都被映射成了表格。其中，选课表中的 `stu_id` 和 `cour_id` 分别是引用学生表和课程表的外键；教课表也是如此。

学生

| <code>stu_id</code> | <code>stu_name</code> | <code>sex</code> | <code>age</code> |
|---------------------|-----------------------|------------------|------------------|
|---------------------|-----------------------|------------------|------------------|

课程

| <code>cour_id</code> | <code>cour_name</code> | <code>xuefen</code> |
|----------------------|------------------------|---------------------|
|----------------------|------------------------|---------------------|

教师

| <code>tea_id</code> | <code>tea_name</code> | <code>sex</code> | <code>age</code> |
|---------------------|-----------------------|------------------|------------------|
|---------------------|-----------------------|------------------|------------------|

选课

| <code>stu_id</code> | <code>cour_id</code> | <code>chengji</code> |
|---------------------|----------------------|----------------------|
|---------------------|----------------------|----------------------|

教课

| <code>tea_id</code> | <code>cour_id</code> |
|---------------------|----------------------|
|---------------------|----------------------|

图 1.14 关系数据模型中的表格

关系模型通过规范化的关系为用户提供了一种简单的用户逻辑结构，具体特征如下。

① 在关系模型中，无论实体还是实体之间的联系都会被映射成统一的关系，即一个二维表。这样的表达方式简单、灵活，使用与维护也很方便。

② 关系型数据库可用于表示实体之间的多对多联系，借助第三个关系表即可，例如，学生选课系统中学生和课程之间表现出一种多对多联系，则需要借助第三个表，也就是选课表将二者联系起来。

③ 关系必须规范化，即每个属性都是不可分割的实体，不允许表中表的存在。

④ 关系模型中的存取路径对用户而言是完全隐蔽的，使程序和数据具有高度的独立性，其数据语言的非过程化程度较高。

但是，关系模型也存在不足之处，具体如下。

➤ 查询效率低。关系数据模型提供了较高的数据独立性和非过程化的查询功能（查询时只需指明数据存在的表和需要的数据所在的列，不用指明具体的查找路径），因此加大了系统的负担。

➤ 由于查询效率较低，因此需要 DBMS 对查询进行优化，加大了 DBMS 的负担。

### 任务 1.3.3 认识关系完整性约束



微课视频

在关系模型中,将能唯一地标识一个元组的属性或属性组称为候选码,一般选中其中一个作为该关系的主码;将包含在任何一个候选码中的属性均称为主属性;将不包含在任何一个候选码中的属性均称为非主属性。

关系模型定义了3种类型的完整性约束条件:实体完整性约束、参照完整性约束和用户定义完整性约束。前两类是关系模型必须满足的完整性约束条件,由关系数据库管理系统自动支持,而最后一类是用户针对特定的数据设置的约束条件。

#### 1. 实体完整性

实体完整性规则:若属性(指一个或一组属性)A是基本关系R的主属性,则属性A不能取空值。

这个规则很容易理解,因为主码可以唯一标识关系中的元组,若构成主码的主属性取空值(所谓空值就是“不知道”或“无意义”的值),则该实体便失去唯一标识功能。例如,关系模型为学生(学号,姓名,性别,年龄,籍贯,专业名称),其中学号是主码,而主码对应的属性只有学号,所以学号也是主属性。根据实体完整性约束规则,学号不能取空值。如果学号取空值,那么这个实体就没有意义了。在学生选课关系模型中,选课(学号,课程号,成绩)中,属性组“学号,课程号”为主码,所以“学号”和“课程号”这两个属性均不能取空值。

实体完整性规则是针对基本关系而言的,即针对现实世界的一个实体集,而现实世界中的实体是可区分的。该规则的目的是利用关系模型中的主码及主属性来区分现实世界中的实体集中的实体,所以主属性不能取空值。

#### 2. 参照完整性

在关系模型中,实体与实体之间的联系采用关系模型来描述。通过引用对应实体的关系模型的主码来表示对应实体之间的联系。

比如,职工关系模型如下。

- 部门(部门编码,部门名称,电话,办公地址)。
- 职工(职工编码,姓名,性别,年龄,籍贯,所属部门编码)。

其中,“所属部门编码”与部门关系模型中的主码“部门编码”相对应,则“所属部门编码”是职工关系模型中的外码。职工关系模型通过外码来描述与部门关系模型的关联。职工关系中的每个元组(每个元组描述一个职工实体)通过外码表示该职工所属的部门。当然,被参照关系的主码和参照关系的外码可以同名,也可以不同名。被参照关系与参照关系可以是不同关系,也可以是同一关系。

比如,职工(职工编码,姓名,性别,年龄,籍贯,所属部门编码,班组长编码),其中的“班组长编码”与本身的主码“职工编码”相对应,属性“班组长编码”是外码,职工关系模型既是参照关系也是被参照关系。

参照完整性规则:若属性F是基本关系R的外码,并且属性F与基本关系S的主码K相对应,则对于基本关系R中每个元组在属性F上的值,必须等于基本关系S中某个元组的主码值,或者取空值。

在职工关系中,如果某一个职工“所属部门编码”取空值,则表示该职工未被分配到指定部门;如果某一个职工“所属部门编码”等于部门关系中某个元组的“部门编码”,则表示该职工隶属于指定部门。如果既不为空值,又不等于被参照关系,则表示该职工被分配到一个不存在的部门,这就违背了参照完整性规则。所以,参照完整性规则就是定义外码与主码之间的引用规则,也是关系模型之间的关联规则。

### 3. 用户定义完整性

用户定义完整性是针对某一具体数据库的约束条件,反映了某一具体应用所涉及的数据必须满足语义要求。关系模型应提供定义和检验这一类完整性的约束机制,以使用统一的系统方法处理它们,而不是由应用程序来提供这一功能。

例如,在职工关系中,职工年龄的取值范围一般限定在16~60,学生选课成绩的取值范围应该限定在0~100,关系模型应该为用户提供定义和检验这一类完整性的约束机制,保证数据的正确性。

## 1.4 任务小结

- **任务 1:** 认识数据库体系结构。通过学习,读者应当对数据、数据库、数据库系统、数据库管理系统等有了基本认识,对数据库体系结构有了比较全面地了解和认识,并了解到即将学习的 MySQL 数据库是 DBMS 的常见品牌之一。
- **任务 2:** 认识数据库的模型。数据模型是数据库系统的核心和基础,本章重点介绍了概念数据模型的 E-R 图表示方法,实体与属性的划分原则,以及逻辑数据模型的层次模型、网状模型和关系模型等 3 种模型特征。通过学习,读者应当对数据模型有了比较全面的了解和认识,并了解到即将学习的 MySQL 数据库是关系型数据库。
- **任务 3:** 认识关系完整性约束。通过学习,读者应当对实体完整性、参照完整性和用户定义完整性 3 种完整性约束有了比较全面地了解,并认识到完整性约束是为了防止数据库中存在不符合语义规定的数据库,以及因错误信息的输入/输出造成无效操作或错误信息而提出的,从而进一步了解到存储在数据库中的所有数据值均需在正确的状态。

通过对本章的学习,读者可以将注意力集中在掌握数据库体系结构相关概念、术语等基本知识方面,为后面章节的学习奠定必备的理论基础。

## 1.5 知识拓展

### 常用的 5 种完整性约束条件

(1) 主键 (Primary Key) 约束: 指定表的一列或几列的组合值在表中具有唯一性,即能唯一指定一行记录。每个表中只能有一列被指定为主键, IMAGE 和 TEXT 类型的列不能被指定为主键,也不允许指定主键列有 NULL 属性。

(2) 外键 (Foreign Key) 约束: 外键定义了表之间的关系,当一个表中的一列或多列的组合和其他表中的主键定义相同时,就可以将这些列或列的组合定义为外键,并设定其

适合将哪个表中的哪些列相关联。这样一来，当在定义主关键字约束的表中更新列值时，其他表中有与之相关联的外关键字约束的表中对应的外关键字列也将进行相同的更新。外关键字约束的作用还体现在：当向含有外关键字的表插入数据时，如果与之相关联的表的列中没有与插入的外关键字列值相同的值时，则系统会拒绝插入数据。与主键相同，不能使用一个定义为 TEXT 或 IMAGE 数据类型的列创建外键。

(3) 唯一性 (Unique) 约束：指定一个或多个列的组合值具有唯一性，以防止在列中输入重复的值。唯一性约束指定的列可以有 NULL 属性。由于主键值是具有唯一性的，因此主键列不能再设定唯一性约束。

(4) 默认值 (Default) 约束：通过定义列的默认值或使用数据库的默认值对象绑定表的列，来指定列的默认值。

(5) 检查 (Check) 约束：对输入列或整个表中的值设置检查条件，以限制输入值，保证数据库的数据完整性。检查约束可以对每个列设置约束检查。

## 1.6 巩固练习

### 一、基础练习

1. 数据库系统一般包括\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_和\_\_\_\_\_ 4 个部分。
2. 根据模型的应用目的，可以将数据库模型可以分为\_\_\_\_\_、\_\_\_\_\_和\_\_\_\_\_。
3. 常用的 5 种完整性约束条件是\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_。
4. 在 E-R 图中，矩形表示\_\_\_\_\_。
5. 用二维表结构表示实体及实体间联系的数据模型被称为\_\_\_\_\_数据模型。
6. 解释概念模型相关术语：实体、实体型、实体集。

### 二、进阶练习

1. 简述数据库系统分别从哪些方面来保证数据的完整性和安全性。
2. 简述网状模型、层次模型、关系模型 3 种模型的特征及优缺点。
3. 简述什么是数据库管理系统，以及它的主要功能有哪些。
4. 设计题：有“商店”和“顾客”两个实体，“商店”属性包括商品编号、商店名、地址、电话，“顾客”属性包括顾客编号、姓名、地址、年龄、性别。假设一个商店有多个顾客购物，一个顾客可以到多个商店购物，顾客每次去商店购物都有一个消费金额和日期，并规定每个顾客在每个商店里一天最多消费一次，请根据上述语义画出 E-R 图。