

1

数据库基础知识



本章导读

数据库基础是初学者了解数据库领域的发展情况、基本概念、特点以及研究方法等基本内容的入门知识。本章主要介绍数据库的基本概念、数据管理的发展概况、信息描述的基本方法、数据模型、关系数据库及其操作、数据库表的设计及其规范化处理方法等内容。读者应在理解相关概念的基础上重点掌握 E-R 模型的设计方法、设计过程以及数据库表的关系规范化处理方法等内容。



本章要点

- 三个世界的信息描述方法
- E-R 模型的设计方法
- 关系数据库的关系操作
- 函数依赖关系、范式
- 关系规范化处理方法

1.1 数据库概述

数据库技术是计算机技术的一个重要分支，计算机应用技术发展到现在，除操作系统之外，数据库已经成为计算机应用的重要支撑系统和核心组成部分。例如，电子政务系统、电子商务系统、决策支持系统、电子邮件系统、财务金融系统等计算机应用系统均以数据库为信息存储的平台，全部都离不开数据库。

1.1.1 数据库基本概念

信息、数据、数据库、数据库管理系统、数据库系统是与数据库技术密切相关的基本概念，理解这些基本概念的含义有助于进一步深入学习和掌握数据库管理系统的应用。

1. 数据 (Data)

数据是对客观事物属性的一种符号化的表示。从数据处理的角度看,数据是计算机处理及数据库中存储的基本对象。数据的表现形式很多,它们都可以经过数字化后存入计算机。例如,数字、字母、文字、图像、声音等在计算机中都以数据的形式体现。

2. 信息 (Information)

信息是经过加工处理并对人类客观行为产生影响的事物属性的表现形式。信息具有实效性、实用性和知识性等特性。

信息与数据是有差别的,任何事物的属性都是通过数据来表示,数据经过加工处理后,便具有了知识性,并对人类活动产生决策作用,从而形成信息;信息与数据是有关系的,信息是数据的内涵,数据是信息的载体。同一条信息可以有不同的数据表示形式,而同一个数据也可以有不同的解释。

3. 数据库 (DB)

数据库是数据存储的仓库,是指长期存储在计算机内、有组织、可共享的数据的集合。数据库中的数据按一定的数据模型进行组织、描述和存储,具有较小的冗余度、较高的数据独立性和易扩展性,并可为各种用户所共享。

4. 数据库管理系统 (DBMS)

数据库管理系统是数据库系统中对数据进行管理的软件系统,是数据库系统的核心组成部分,它为用户提供一个可以方便、有效地存取数据库信息的环境。数据库管理系统的功能由数据定义、数据库的运行管理、数据库的建立和维护、数据操纵等四个方面组成。

5. 数据库系统 (DBS)

数据库系统是采用数据库技术构建的复杂的计算机系统,它是综合了计算机硬件、软件、数据集合和数据库管理人员、遵循数据库规则、向用户和应用程序提供信息服务的集成系统。数据库系统由数据库、软件系统、硬件系统、数据库管理系统、数据库管理员和用户等要素组成。

1.1.2 数据库的发展

数据库技术是由于数据管理任务的需要而产生的,数据管理技术发展至今大致可分为人工管理、文件管理、数据库系统管理等三个阶段。

1. 人工管理阶段

人工管理阶段出现在计算机应用于数据管理的初前期。计算机没有问世之前,文件管理主要是使用纸张等媒介对数据进行管理;计算机问世初期,由于计算机的软、硬件技术不像今天那么先进,用户的应用程序与一组数据直接对应,程序不仅要设计数据处理的方法,还要直接操作数据。在这种管理方式下,用户的应用程序与数据之间相互结合、不可分割,当数据有所变动时程序也必须随之改变,因此,程序与数据的独立性差,程序之间的数据不能互相传递,缺少共享性。

2. 文件管理阶段

文件管理阶段是把有关的数据组织成一种数据文件,数据可以脱离程序独立存在。在这种管理方式下,用户或应用程序通过文件管理系统对数据文件中的数据进行加工处理,应用程序的数据具有一定的独立性,比手工管理方式前进了一步。但是,数据文件仍然高度依赖于其应用程序,不能被多个程序所共享。

3. 数据库管理阶段

随着数据量的不断增大,文件管理方式显然在操作效率和数据共享方面不能满足实际要求,为

了解决在海量数据、数据共享和操作效率等方面存在的问题，于是出现了数据库。

数据库管理阶段对所有的数据实行统一的规划管理，所有数据形成一个数据中心，构成一个数据仓库。在这种管理方式下，应用程序不再只与孤立的数据文件相对应，而是能够从整体数据集中取出某个子集作为逻辑文件与应用程序相对应，并通过数据库管理系统实现逻辑文件与物理数据之间的映射。在数据库管理的系统环境下，应用程序对数据的管理和访问灵活方便，数据与应用程序之间相互独立，使得程序的编制质量和效率大大提高，数据的冗余度极大地减少，数据的共享性得到了显著的增强。

1.2 数据模型

数据库管理系统是基于某个数据模型设计出来的，而数据模型是实现现实世界数据特征的模拟和抽象。一个完整的数据模型主要包括数据结构、数据操作和数据完整性约束等三个部分。其中，数据结构用来描述实体之间的构成和联系；数据操作是指对数据库的查询和更新操作；数据完整性约束则是指施加在数据上的限制和规则，目的是使数据库中的数据更具可用性和有效性。

数据模型的设计要满足三方面的要求：一是能够真实地模拟现实世界；二是容易理解；三是能够在计算机上实现。

根据应用目的的不同，数据模型分为两种：一种是信息模型，也称概念模型，它从用户的观点来对数据和信息进行建模，反映了信息从现实世界到信息世界的转化，它不涉及计算机软、硬件的具体细节，只注重于符号表达和用户的理解能力。典型的信息模型有著名的“E-R（实体-联系）”模型。信息模型主要用于数据库设计阶段。另一种是结构数据模型，结构数据模型主要用于 DBMS 的实现，它反映了信息从信息世界到机器世界的转换，描述了计算机中数据的逻辑结构、信息在存储器上的具体组织等。常见的结构数据模型有三种，即层次模型、网状模型和关系模型。

1.2.1 数据处理的抽象描述

人们在研究和处理数据的时候，通常把数据的描述分为三个世界，即现实世界、信息世界、机器世界，这三个世界对信息描述的转换过程，就是将客观现实的信息反映到计算机数据库中的过程。

1. 现实世界

客观存在的世界就是现实世界，它独立于人们的思想之外。现实世界存在无数的对象和事务，每一个对象或事务可以看成是一个个体，每个个体有一项或多项特征信息。例如，把人看成对象时，有身高、体重、年龄、肤色等基本特征。

2. 信息世界

信息世界是现实世界在人们头脑中的反映，人的思维将现实世界中对象或事务的特征抽象化后用文字符号表示出来，就形成了信息世界。描述信息世界的常用术语有：

(1) 实体：客观存在的并且可以相互区别的事物称为实体，实体可以是具体的事物，也可以是抽象的事件。例如，一名学生、一台电脑、上课、比赛等。

实体是信息世界的基本单位，相同类型的实体的集合称为实体集。例如，一个班由多位学生实体组成，则这个班集体称为学生实体的实体集。

(2) 属性：实体的特性称为属性，一个实体可以有多个属性，每一个属性都有其数据类型和数据的取值范围。例如，学生实体可由学号、姓名、成绩等若干属性来描述。

(3) 键：在一个实体集中能唯一标识一个实体的属性称为键。键可以是一个属性，也可以是多个属性的组合。例如，学号、身份证号等能唯一地标识一个学生，它们都是键。

(4) 联系：实体之间相互作用、互相制约的关系称为实体集的联系，也称为关联或关联关系。实体之间的关联关系主要有一对一、一对多（多对一）和多对多等三种。

3. 机器世界

机器世界又称为数据世界。信息世界中的信息经过抽象和组织，以数据的形式存储在计算机中，从而形成了机器世界。在机器世界中，用于描述数据的基本术语与信息世界中的术语是一一对应的，只是在文字叙述上略有差异。

(1) 字段：字段也称数据项，用于标记实体的属性。在一张关系表中，每一列称为一个字段。字段与信息世界的“属性”相对应。例如，在学生信息表中，一个学生就是一个实体，它包含了学号、姓名、性别、年龄等字段。

(2) 记录：记录是具有逻辑关系的一个或多个字段的集合。记录与信息世界中的“实体”相对应，一个记录描述了一个实体的基本信息。

(3) 关键字：能够唯一标识一条记录的字段称为关键字。关键字与信息世界中的“键”相对应。关键字可以是一个字段，也可以由多个字段组合而成。例如，学生信息表中的学号、身份证号等能唯一地标识一名学生，它们都是关键字。

(4) 文件：文件是记录的集合。文件对应于信息世界中的“实体集”。文件的存储形式有很多种，例如，顺序文件、随机文件、索引文件等。

4. 三个世界信息描述的对应关系

从现实世界到信息世界再到机器世界，事务被逐层抽象，信息被逻辑化、符号化等处理，表 1.1 展示了三种世界之间信息描述的对应关系。

表 1.1 三个世界信息描述的对应关系

现实世界	信息世界	机器世界
特征	属性	字段
唯一特征	键	关键字
事务	实体	记录
	实体集	文件

1.2.2 实体-联系模型

实体-联系模型（简称 E-R 模型）是目前最常用的信息模型，它从用户的观点来对数据和信息进行建模，反映了信息从现实世界到信息世界的转化，它不涉及计算机软、硬件的具体细节，便于分析与理解。

1. E-R 图的基本元素

E-R 模型使用 E-R 图来表达，E-R 图主要由实体、属性、联系和连线等元素符号组成。各元素的表达符号和用途如表 1.2 所示。

2. E-R 图的设计

完整的 E-R 图必须清楚地表达实体、属性和联系三者之间的关系。下面根据常见的关联关系的种类分别介绍 E-R 图的设计方法。

表 1.2 E-R 图的基本元素和用途

名称	符号	符号用途
矩形框	实体	表示实体，框内写实体的名称
椭圆框	属性	表示实体的属性，框内写属性的名称，该属性是实体的非主键
粗边椭圆框	键	表示实体的属性，框内写属性的名称，该属性是实体的主键
线条	—	用于实体、属性、键、联系框之间的连接
菱形框	联系	表示实体间的联系，框内写联系的名称。旁边标上关联关系的类型

(1) 一对一关联关系

如果实体集 A 中的每一个实体，在实体集 B 中最多只能有一个实体与之有关系，反之亦然，则称实体集 A 与实体集 B 的关系是一对一的关联关系，记为 1:1。

【例 1-1】使用 E-R 图描述学校和校长两个实体的关联关系。

【案例分析】因为一个学校只有一个校长，而一个校长只在一个学校任职。所以，学校和校长的关系是一对一的关联关系。设学校实体的属性主要有：编号、校名、地址等；校长实体的属性主要有：编号、姓名、职称等。使用 E-R 图描述如图 1-1 所示。

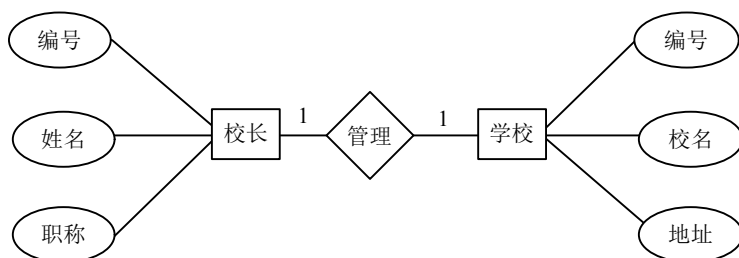


图 1-1 一对一关联关系

(2) 一对多或多对一关联关系

如果实体集 A 中的每一个实体，实体集 B 中有 n 个实体 ($n \geq 0$) 与之有关系，则称实体集 A 与实体集 B 是一对多的关联关系，记为 1:n。反之，如果实体集 A 中有 n 个实体 ($n \geq 0$) 与实体集 B 中的一个实体有关系，则称实体集 A 与实体集 B 是多对一的关联关系，记为 n:1。

一对多与多对一的关联关系是同一种关系，只是观察问题的方向不同而已。

【例 1-2】使用 E-R 图描述班级与学生两个实体的关联关系。

【案例分析】因为一个班有多名学生，而每个学生只能是某个班级的成员，所以，班级与学生的关系是一对多的关联关系。设班级实体的主要属性有：编号、班名、教室等；学生实体的主要属性有：编号、姓名、年龄等。使用 E-R 图描述如图 1-2 所示。

(3) 多对多关联关系

如果实体集 A 中的每一个实体，实体集 B 中有 n 个实体 ($n \geq 0$) 与之有关系。反之，如果实体集 B 中的每一个实体，实体集 A 中也有 m 个实体 ($m \geq 0$) 与之有关系，则称实体集 A 与实体集 B 是多对多的关联关系，记为 m:n。

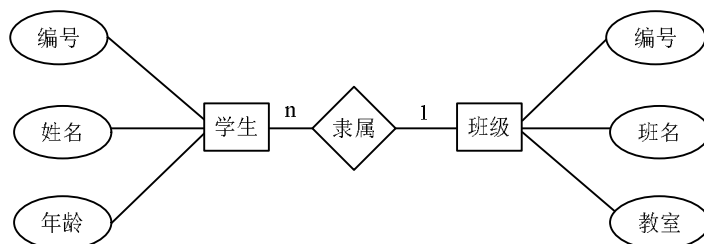


图 1-2 一对多或多对一关联关系

【例 1-3】使用 E-R 图描述课程与学生两个实体的关联关系。

【案例分析】因为一门课程同时有若干个学生选修，而一个学生可以同时选修多门课程。所以，课程与学生之间的关系是多对多的关联关系。设课程实体的主要属性有：编号、名称、学分等；学生实体的主要属性有：编号、姓名、年龄等。使用 E-R 图描述如图 1-3 所示。

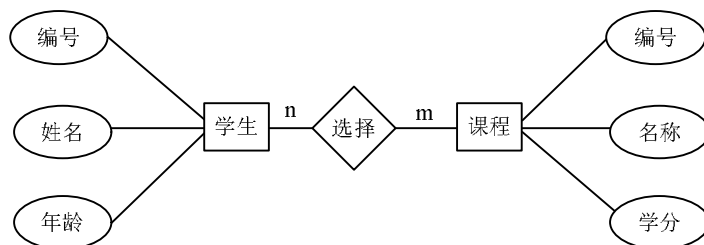


图 1-3 多对多关联关系

(4) 实体集自身发生的关联关系

如果实体集 A 中的某个实体与实体集 A 中的另一个实体产生关系，则称实体集 A 自身具有一对一的关联关系；如果实体集 A 中的某个实体与实体集 A 中的 n 个实体 ($n \geq 0$) 产生关系，则称实体集 A 自身具有一对多的关联关系。

【例 1-4】使用 E-R 图描述应用程序中菜单实体的主菜单与子菜单项的关联关系。

【案例分析】在菜单实体集中，主菜单与子菜单都是菜单集的实体，但是，一个主菜单通常有 n 个 ($n \geq 0$) 子菜单实体与之产生关系，所以，主菜单与子菜单的关系是实体集中自身发生的一对多的关联关系。设菜单实体的属性有：编号、名称、类型、动作、关联等。使用 E-R 图描述主菜单与子菜单两个实体之间的关联关系时，如图 1-4 所示。

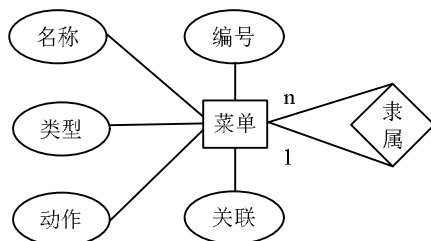


图 1-4 自身发生的一对多关联关系

(5) 多个实体之间发生的关联关系

如果实体集 A 与实体集 B 发生某种关系（如一对多等），而实体集 B 与实体集 C 也发生某种

关系（如一对多、一对一或多对多等），则多个实体之间可能两两产生关联关系，这是实际应用中常见的综合性关联关系。

【例 1-5】使用 E-R 图描述学生、课程和教师三个实体之间的关联关系。

【案例分析】学生实体和课程实体的关联关系由例 1-3 所述。教师实体与课程实体的关系是一个教师可以授 n 门 ($n \geq 0$) 课程，所以，教师实体与课程实体是一对多的关联关系。设教师实体的主要属性有：编号、姓名、职称等。使用 E-R 图描述学生、课程和教师三个实体之间的关联关系时，如图 1-5 所示。

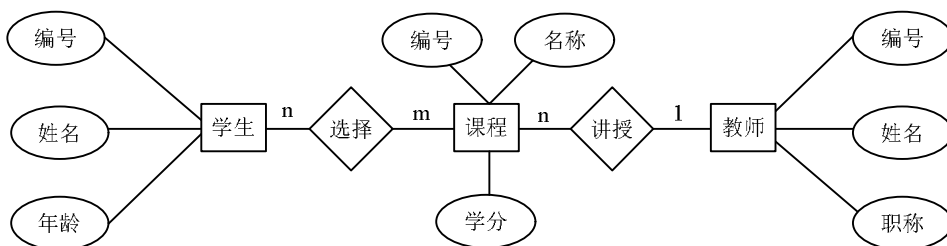


图 1-5 多个实体之间的关联关系

在一个应用系统中，把所有实体间产生的关系利用 E-R 图描述出来并组合成一个总的 E-R 图后，就得到了该应用系统的完整的 E-R 模型图。

设计 E-R 模型时，由于设计人员的经验或见解不同，针对同一个企业的运营模式，不同人员设计的 E-R 模型是有差异的，因此，E-R 图不是唯一的。在实际应用中，当 E-R 图比较大时，为了使 E-R 图表达更抽象，可以将属性省略，只保留实体和联系两部分内容。

总之，使用 E-R 图表达的 E-R 模型直观易懂，它是系统开发人员和客户沟通的良好工具。对客户来说，它概括了企业运营的方式和各种联系，对系统开发人员来说，它从概念上描述了一个应用系统数据库的信息组织结构。根据 E-R 图，结合具体的 DBMS 可以设计出相应的数据库管理系统的结构数据模型。因此设计完善、高质量的 E-R 图是数据库设计的一个重要步骤。

1.2.3 结构数据模型

结构数据模型是机器世界的数据库模型，常见的结构数据模型有层次模型、网状模型以及关系模型。采用这些模型构建的数据库管理系统分别叫做层次型、网状型以及关系型数据库管理系统。

1. 层次模型

以树型结构表示实体之间联系的模型称为层次模型。在这种模型中，数据被组织成由“根”开始的“树”，每个实体由根节点开始沿着不同的分支放在不同的层次上。如果不再向下分支，则分支序列中最后的节点称为“叶”节点。图 1-6 是一个层次模型的实例。

层次模型的主要特征是：有且仅有一个节点无父节点，这个节点称为根节点；除根节点外，每一个节点有且只有一个父节点；父节点与子节点之间是一对多的关联关系；层次模型无法表达多对多的关联关系。

2. 网状模型

用网状结构表示实体之间联系的模型称为网状模型。网状模型是层次模型的扩展，网中的每一个节点代表一个实体类型。网状模型突破了层次模型的两点限制：允许节点有多于一个的父节点；

可以有一个以上的节点没有父节点。图 1-7 是一个网状模型的实例。

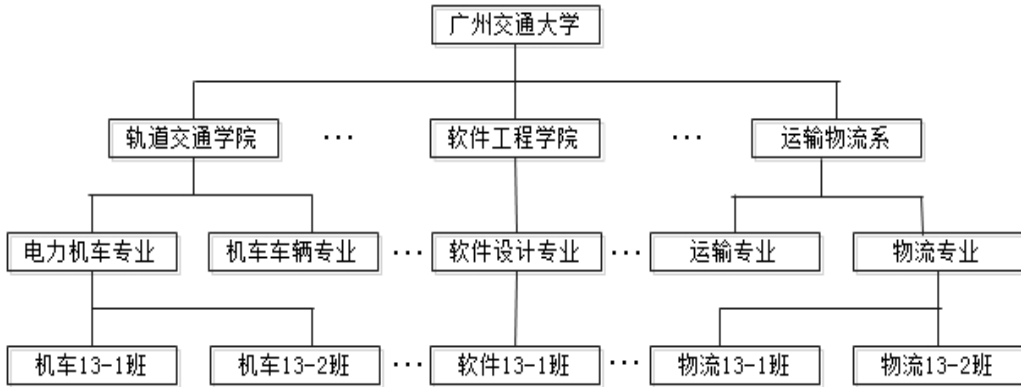


图 1-6 层次模型实例

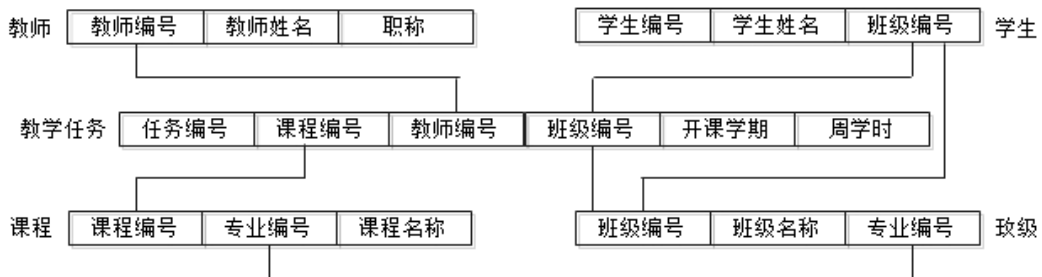


图 1-7 网状模型实例

网状模型的主要特征是：允许一个以上的节点没有父节点；一个节点可以有多个父节点；节点间是多对多的关系。

3. 关系模型

采用二维表的结构表示实体之间的联系的模式称为关系模型。关系模型以关系数学理论为基础，操作的对象和结果都是二维表，这种二维表与日常生活、工作中所使用的行列表格的概念完全一样，所以，关系模型是目前应用最多、使用最广泛的数据模型。

在关系模型中，无论实体本身还是实体间的联系均用称为“关系”的二维表来表示。表 1.3 列出了学生入学时的概要信息，这是一张典型的二维关系表格，表中各行的数据从层次隶属的角度看是没有关系的，即表中的行数据虽然是平行的，但不代表从属关系。

表 1.3 学生入学概要信息表

学号	姓名	性别	出生日期	入学总分	入学时间	班级关联号 class_id
0101110101	李表华	女	1995-01-01	504	2011-09-01	1
0101110102	陈小东	男	1994-09-24	486	2011-09-01	1
0101120103	李明明	女	1995-11-26	483	2012-09-01	3

关系模型概念清晰、结构简单，用户比较容易理解，具有较好的数据独立性和安全保密性。但是，关系模型也有缺点，其查询效率有时不如层次模型和网状模型。

1.3 关系数据库

关系模型是当今最流行的数据模型，较好地解决了网状模型和层次模型中存在的问题，目前几乎所有流行的数据库管理系统产品如 Oracle、SQL Server、MySQL 等都是基于关系模型的实现。

1.3.1 关系数据库的基本概念

关系数据库是基于关系模型建立的数据库。关系数据库由一个或多个称为“关系”的二维表组成，每个关系又由一条或多条记录组成，而每条记录则由一个或多个称为“字段”的属性组成。在关系数据库中，大部分基本概念与描述信息世界的常用术语相一致。

1. 关系

关系是指一张有规则的、没有重复行或重复列的二维表。一个关系对应于一张二维表，每张二维表（下面常用“关系”简称）有一个表名。

2. 记录

在关系中，水平方向的行称为记录。记录也称为实体或元组（下面常用“记录”简称）。

3. 属性

在关系中，垂直方向的列称为属性。属性在关系中也叫列、栏、字段等（下面常用“字段”简称），每一列有一个属性名，也叫列名、栏名或字段名等（下面常用“字段名”简称）。关系中的字段是有值的，它等于当前被处理的记录所对应的字段值。

4. 域

字段的取值范围称为域。例如，“性别”字段只能取“男”或“女”两者之一。

5. 关联

关联是指关系与关系之间记录数据相互关联的方式。关联也称联系或关联关系（下面常用“关联关系”简称）。关联关系是数据库规范化处理时需要重点解决与优化处理的核心问题之一，主要有一对一、一对多（多对一）、多对多等三种。

6. 关键字

在关系中，能够唯一标识一条记录的字段或多个字段的组合称为关键字。关键字也称为“键”。例如，在表 1.3 中，“学号”能唯一标识一名学生，可以充当关键字；但学生的“姓名”不能拿来充当关键字，因为同一个班级可能出现两个或多个同名同姓的学生。

7. 候选关键字

当关系中存在两个或多个“键”时，称这些“键”为候选关键字或候选键。

8. 主关键字

在关系中，从一个或多个候选关键字中选出一个关键字做主关键字，则这个关键字称为主关键字，也称主键（下面常用“主键”简称）。在一个关系中主键只能有一个。

9. 外部关键字

在关系中，如果存在这样一个字段：一方面，它不是本关系的关键字，另一方面，它的取值来自另一个关系的某个主键，则此字段称为外部关键字，也叫外键。

1.3.2 关系的特点

- (1) 关系中的字段不可再分，必须是最基本的原子数据项。
- (2) 关系中每一字段的值的数据类型相同，并且来自同一个域。
- (3) 字段在关系中的左、右次序是无关系的。
- (4) 记录在关系中的上、下次序是无关系的。
- (5) 同一关系中不允许出现两个相同的记录。
- (6) 同一关系中不允许出现两个相同的字段。

1.3.3 数据完整性

数据完整性是为保证数据库中关系数据的正确性和相容性,对关系模型提出的约束条件和需要遵守的规则。数据完整性通常包括实体完整性、域完整性和参照完整性。

1. 实体完整性

实体完整性是在关系内部建立的记录与记录之间的约束规则。它通过关系的主键来实现约束,要求关系中主键的值不能为“空”,并保证数据具有唯一性,这样就能确保关系中不会出现重复的记录。实体完整性解决关系内部记录与记录之间数据的约束问题。

2. 域完整性

域完整性是指在关系内部建立的对字段的定义和取值范围进行限制的约束规则。域完整性解决的是字段自身的数据约束问题。

3. 参照完整性

参照完整性是指在关系与关系之间(或在同一个关系的不同记录之间)建立的约束规则。它通过一个关系的外键与另一个关系的主键建立起关联关系来进行关系之间的约束。参照完整性主要用于解决关系与关系之间记录数据的关联约束问题。

1.3.4 关系操作

关系数据库的操作主要有选择、投影和连接三种。数学中的集合运算是关系操作的理论基础。

1. 选择操作

选择操作也叫筛选操作,是指从指定的关系中选择出满足某种条件的记录从而形成一个新的关系的操作。

【例 1-6】从“class 班级表”(见图 1-8)中选择出“2012”年级的班级信息。

如图 1-9 所示的记录集是从图 1-8 所示的班级表中选择出来的“2012”年级的班级信息,从结果可知,选择操作的结果是被选择的关系的子集。

2. 投影操作

投影操作是对指定的关系进行字段的投影,它从指定的关系中抽取出某些满足需要的字段列以形成一个新的关系,为此,投影操作可以看成是对某个关系进行列的选择操作,

【例 1-7】从“class 班级表”中投影出“班级编号、班级名称”两个字段的的信息。

如图 1-10 所示的记录集是从图 1-8 所示的班级表中投影所得,从结果可知,投影操作的结果是被投影关系的字段的子集。

id	班级编号	班级名称	年级	specialty_id
1	01011101	计算机11-1	2011	1
2	01011102	计算机11-2	2011	1
3	01011201	计算机12-1	2012	1
4	01011202	计算机12-2	2012	1
5	01021101	网络11-1	2011	2
6	01021201	网络12-1	2012	2
7	01031101	多媒体11-1	2011	3
8	01031201	多媒体12-1	2012	3
9	01041101	动漫11-1	2011	4
10	01041201	动漫12-1	2012	4

图 1-8 class 班级表

id	班级编号	班级名称	年级	specialty_id
3	01011201	计算机12-1	2012	1
4	01011202	计算机12-2	2012	1
6	01021201	网络12-1	2012	2
8	01031201	多媒体12-1	2012	3
10	01041201	动漫12-1	2012	4

图 1-9 选择操作

班级编号	班级名称
1	01011101 计算机11-1
2	01011102 计算机11-2
3	01011201 计算机12-1
4	01011202 计算机12-2
5	01021101 网络11-1
6	01021201 网络12-1
7	01031101 多媒体11-1
8	01031201 多媒体12-1
9	01041101 动漫11-1
10	01041201 动漫12-1

图 1-10 投影操作

3. 连接操作

连接操作是从两个或多个关系的记录中选择出满足某一给定条件的记录作为操作结果。在连接操作中，一般都包含选择操作和投影操作，如果包含选择操作则会改变结果关系的记录条数，如果包含投影操作则会改变结果关系的字段个数。

两个关系进行连接操作后，如果连接操作没有包含投影操作，则结果关系的字段个数是两个关系的字段的并集；如果连接操作没有包含选择操作，则结果关系的记录条数是两个关系记录的笛卡尔乘积。例如，设两个关系的记录数分别为 N 和 M ，则笛卡尔乘积的记录数为 $N \times M$ 。

【例 1-8】通过连接操作从“class 班级表”和“specialty 专业表”中产生“计算机应用技术”专业的班级专业信息：班级编号、班级名称、专业名称。

【案例分析】根据案例的要求，图 1-8 所示的班级表和图 1-11 所示的专业表进行连接操作时，首先要进行关系的连接操作，然后进行选择操作，最后进行投影操作才能得到结果。

id	专业编号	专业名称
1	0101	计算机应用技术
2	0102	计算机网络技术
3	0103	计算机多媒体技术
4	0104	动漫设计与制作

图 1-11 specialty 专业表

操作过程如下：

第一步：两表要根据关联关系的条件，即“class 班级表”的外键“specialty_id”等于“specialty 专业表”的主键“id”，将两表的记录建立起连接关系，结果如图 1-12 所示。

id	班级编号	班级名称	年级	specialty_id	id	专业编号	专业名称	
1	1	01011101	计算机11-1	2011	1	1	0101	计算机应用技术
2	2	01011102	计算机11-2	2011	1	1	0101	计算机应用技术
3	3	01011201	计算机12-1	2012	1	1	0101	计算机应用技术
4	4	01011202	计算机12-2	2012	1	1	0101	计算机应用技术
5	5	01021101	网络11-1	2011	2	2	0102	计算机网络技术
6	6	01021201	网络12-1	2012	2	2	0102	计算机网络技术
7	7	01031101	多媒体11-1	2011	3	3	0103	计算机多媒体技术
8	8	01031201	多媒体12-1	2012	3	3	0103	计算机多媒体技术
9	9	01041101	动漫11-1	2011	4	4	0104	动漫设计与制作
10	10	01041201	动漫12-1	2012	4	4	0104	动漫设计与制作

图 1-12 关联结果

第二步：从图 1-12 所示的连接结果中进行选择操作，选择的条件是“专业名称”等于“计算机应用技术”，选择的结果如图 1-13 所示。

第三步：从图 1-13 所示的选择结果中进行投影操作，根据案例要求，投影只抽取班级编号、

班级名称、专业名称三个字段的的信息。投影的结果如图 1-14 所示，这也是案例要求的最终结果。

id	班级编号	班级名称	年级	specialty_id	id	专业编号	专业名称	
1	1	01011101	计算机11-1	2011	1	1	0101	计算机应用技术
2	2	01011102	计算机11-2	2011	1	1	0101	计算机应用技术
3	3	01011201	计算机12-1	2012	1	1	0101	计算机应用技术
4	4	01011202	计算机12-2	2012	1	1	0101	计算机应用技术

图 1-13 选择结果

	班级编号	班级名称	专业名称
1	01011101	计算机11-1	计算机应用技术
2	01011102	计算机11-2	计算机应用技术
3	01011201	计算机12-1	计算机应用技术
4	01011202	计算机12-2	计算机应用技术

图 1-14 投影结果

1.4 关系规范化

一个数据库常常包含多个关系，为了使数据库中关系的结构更加合理，消除各种操作异常，使数据冗余尽可能地小，便于插入、更新和删除操作，数据库中的关系在设计时一般都要经过规范化处理。关系规范化的目的是使得关系模式中的每个属性都是不可再分的数据项、使关系模式尽可能地满足第三甚至第五范式的基本要求。

1.4.1 关系规范化的必要性

下面通过分析表 1.4 的数据结构来理解关系规范化的必要性。表中的数据以学号排序，列出了每位学生的专业、班级、各门课程的授课教师和课程成绩等信息。从学生的角度看，课程的成绩信息比较详尽，但从数据库管理的角度去分析，这个关系却存在诸多问题。

表 1.4 综合信息表

学号	姓名	性别	专业	教师编号	教师姓名	职称	课程编号	课程名称	成绩
0101	张晓青	女	软件(JAVA)	0001	李光策	讲师	J001	软件工程	90
0101	张晓青	女	软件(JAVA)	0002	孙静祥	教授	J002	数据结构	86
0102	王天明	男	软件(JAVA)	0001	李光策	讲师	J001	软件工程	90
0102	王天明	男	软件(JAVA)	0002	孙静祥	教授	J002	数据结构	76

1. 存在非原子属性

首先，关系表中的“专业”属性不是“原子”元素，还可以继续分解成“专业名称”和“专业方向”两个属性。

2. 存在大量的数据冗余

例如，关系中每个教师有教师编号、教师姓名、职称三个数据项，如果一个班有 50 个学生，则每门课程的教师信息将有 50 份相同的数据，存在大量的数据冗余。

3. 容易造成插入、修改和删除异常

从关系中插入、修改和删除数据时，只要数据满足“域”完整性规则，操作就不会产生异常，但容易造成数据不一致。例如，修改某位教师的“职称”数据时，可能会漏改或错改了其他行该教师的职称数据。如果深入分析，该关系还容易产生插入、修改和删除等操作异常。

提示：限于篇幅，关于操作异常这里不作深入探讨，有兴趣的读者可参考数据库设计原理方面的资料或书籍。

通过上述分析，可以确认表 1.4 是一个不好的关系。一个好的关系，不应存在非原子属性、不

应发生插入异常或删除异常、应该尽可能地避免出现修改异常、冗余数据必须尽可能地减少。如何解决这些问题？这正是关系规范化要解决的问题。

关系设计主要包括函数依赖、关系范式和模式设计等三方面的内容，其中，函数依赖起着核心作用，下面作简要介绍。

1.4.2 函数依赖关系

关系的规范化主要是处理关系中的属性对其他某个或某些属性数据的依赖程度，这些依赖条件主要是函数依赖条件。在概念上，函数依赖是指一个或一组属性的取值可以决定其他属性的取值。

常见的函数依赖有完全函数依赖、部分函数依赖、传递函数依赖三种。

在介绍函数的依赖关系之前，下面给出关系的另一种表示方法，即“关系模式”法：

格式：关系名(属性名[, ...n])

例如：上述图 1-11 所示“专业表”使用关系模式可表示为：

专业(id, 专业编号, 专业名称)

1. 函数依赖定义

设 R 是关系模式中的一个关系， X 、 Y 是 R 的属性子集。如果 X 的值确定后，属性集 Y 的值也同时被确定，则称 X 函数决定 Y ，也就是说 Y 函数依赖于 X ，记为： $X \rightarrow Y$ ；如果可逆则为互相依赖，记为： $X \leftrightarrow Y$ ；如果不存在函数依赖，则记为： $X \not\rightarrow Y$ 。

例如，设有关系“学生(学号, 姓名, 性别)”，则属性“姓名, 性别”都函数依赖于“学号”，即有“学号 \rightarrow 姓名, 学号 \rightarrow 性别”，换句话说，“学号”可以决定“姓名, 性别”等属性的取值。

2. 完全函数依赖

如果 Y 属性集依赖于 X ，而不依赖于 X 的任何一个真子集，则称 Y 完全函数依赖于 X ，记为： $X \xrightarrow{f} Y$ 。

例如，设有关系“成绩(学号, 课程号, 成绩)”，因为“学号”或“课程号”均不能单独确定一个学生某门课程的成绩，只有两者联合起来才能确定，所以“(学号, 课程号) \xrightarrow{f} 成绩”是完全函数依赖。

关系模式中，如果仅存在一个主键，则非主属性（不属于主键属性集中的属性）都完全函数依赖于主键。

3. 部分函数依赖

设 X' 是 X 的某一个特定的子集，如果 $X \rightarrow Y$, $X' \rightarrow Y$ ，则称 Y 部分函数依赖于 X ，记为： $X \xrightarrow{p} Y$ 。

例如，设有关系“学生(学号, 姓名, 性别, 班级编号, 班级名称)”，因为有(学号, 班级编号) \rightarrow 班级名称, 班级编号 \rightarrow 班级名称，所以“(学号, 班级编号) \xrightarrow{p} 班级名称”是部分函数依赖。

关系模式中存在部分函数依赖是因为某些属性既完全函数依赖于组合键，又完全函数依赖于组合键中的某个键。

4. 传递函数依赖

如果 $X \rightarrow Y$ ，而且 $Y \rightarrow Z$ ，但 $Y \not\rightarrow X$ ，即 Y 不完全函数依赖（部分函数依赖）于 X ，则称 Z 传递函数依赖于 X ，记为： $X \xrightarrow{t} Z$ 。

例如，设有关系“教学(课程号, 教师号, 教师电话)”，因为有“课程号 \rightarrow 教师号, 教师号 \rightarrow 教师电话, 教师号 $\not\rightarrow$ 课程号”，所以“课程号 \xrightarrow{t} 教师电话”是传递函数依赖。

关系模式中如果存在两类或两类以上不同性质的属性集，则关系模式必然存在传递函数依赖。

1.4.3 关系规范化处理

根据一个关系满足数据依赖程度的不同，关系可规范化为第一范式（1NF）、第二范式（2NF）、第三范式（3NF）、……、第五范式（5NF）等。实际应用时，通常只规范化到第三范式即能满足基本要求。

1. 第一范式（1NF）

定义：如果关系模式中的每个属性都是不可分解的基本数据项（简称原子属性），则称这个关系是满足第一范式的关系。第一范式是最低级别的范式。

【例 1-9】将表 1.4 中的关系规范化以满足第一范式的要求。

【案例分析】因为，“专业”属性包含了“专业方向”，所以该关系不满足第一范式的要求，如果将“专业”属性分解为“专业名称、专业方向”两个属性，就满足了第一范式的要求。为便于后面的第二、第三范式的规范化，表 1.4 改用关系模式表示如下：

综合信息表（学号，姓名，性别，专业名称，专业方向，教师编号，教师姓名，职称，课程编号，课程名称，成绩）

从关系模式可以看出，尽管关系满足第一范式的要求，但没有解决数据冗余量大以及操作异常等问题，所以，需要引入第二、第三范式的处理。

2. 第二范式（2NF）

定义：如果关系模式中的所有非主属性都完全函数依赖于任意的候选键，则称这个关系是满足第二范式的关系。第二范式要求关系在满足第一范式的基础上，去除那些部分函数依赖于主键的非主属性，使得关系的所有非主属性都完全函数依赖于主键。

如果关系中出现非主属性对主键的部分函数依赖，则关系就不满足第二范式，所以，规范化处理时，须从消除非主属性对主键属性的部分函数依赖入手。规范化的方法是，对关系进行“投影”处理，将关系模式分解为两个或多个新的关系模式。

【例 1-10】将例 1-9 得到的关系模式规范化以满足第二范式的要求。

【案例分析】分析“综合信息表”关系模式，因为有“(学号，教师编号) → (教师姓名、职称)”，“教师编号 → (教师姓名，职称)”，所以，存在部分函数依赖；类似的，关系模式中还存在专业、课程等信息的部分函数依赖，所以关系模式不满足第二范式的要求。

根据消除部分函数依赖的操作，原有关系可拆分成四个关系：

- (1) 学生（学号，姓名，性别，专业名称，专业方向）。
- (2) 教师（教师编号，教师姓名，职称）。
- (3) 课程（课程编号，课程名称）。
- (4) 成绩（学号，课程编号，教师编号，成绩）。

关系模式分解后，消除了部分函数依赖，达到了第二范式的要求，但是，有些新的关系模式仍然因为存在“传递函数依赖”而导致操作异常，所以，还要继续进行第三范式的规范化处理。

3. 第三范式（3NF）

定义：如果关系模式中的所有非主属性对任何候选键都不存在传递依赖，则称这个关系是满足第三范式的关系。第三范式要求关系在满足第二范式的基本上，除去所有传递函数依赖于主键的非主属性。

【例 1-11】将例 1-10 得到的关系模式规范化以满足第三范式的要求。

【案例分析】因为满足第三范式的关系模式一定满足第二范式的要求，所以第三范式的规范化处理可以从第二范式的规范化结果开始。分析“学生（学号，姓名，性别，专业名称，专业方向）”关系模式，因为“学号 \rightarrow 专业名称，专业名称 \rightarrow 学号，专业名称 \rightarrow 专业方向”，所以，“学号 \xrightarrow{t} 专业方向”出现传递函数依赖。这时，仍然需要将关系再次分解以消去传递函数依赖。

根据第三范式的要求，上述“学生”关系模式可进一步分解为如下三个关系模式：

- (1) 学生（学号，姓名，性别，方向编号）。
- (2) 方向（方向编号，方向名称，专业编号）。
- (3) 专业（专业编号，专业名称）。

至此，“综合信息表”关系模式规范化完毕，得到六个满足第三范式要求的数据库模式：

- (1) 学生（学号，姓名，性别，方向编号）。
- (2) 方向（方向编号，方向名称，专业编号）。
- (3) 专业（专业编号，专业名称）。
- (4) 教师（教师编号，教师姓名，职称）。
- (5) 课程（课程编号，课程名称）。
- (6) 成绩（学号，课程编号，教师编号，成绩）。

关系规范化还可以进行到第四、第五等范式，但不是规范化程度越高越好，因为关系规范化会带来其他一些问题，例如，规范化程度越高，关系模式的表现力就越差、数据操纵就会越复杂、操纵的代价就会越高等。因此，有时出于考虑编程效率等方面的原因，关系模式会人为地加入一些其他属性，使关系只满足第二范式的要求。

小结

(1) 数据模型分为两种，一是信息模型，二是结构数据模型。常见的结构数据模型有层次模型、网状模型和关系模型三种。一个完整的数据模型通常包括数据结构、数据操作和数据完整性约束等三个部分。当前流行的关系数据库管理系统都是关系模型的具体实现。

(2) E-R 模型是目前最常用的信息模型，它从用户的观点来对数据和信息进行建模。由于系统分析人员的经验或见解不同，同一个企业运营模式，不同人员设计出来的 E-R 模型有所差异，因此，E-R 模型设计不是唯一的。

(3) 关系数据库的操作主要有选择、投影和连接三种。其中，选择操作用于从关系中选择满足条件的记录行；投影操作用于从关系中选择满足需要的列；连接操作可以将两个或多个表进行拼接，它通常包含选择和投影两种操作。

(4) 关系规范化时，第一范式的规范化使得关系模式中的每个属性都是不可分解的基本数据项；第二范式的规范化是去除那些部分函数依赖于主键的非主属性，使得关系模式中的所有非主属性都完全函数依赖于主键；第三范式的规范化是除去关系模式中那些传递函数依赖于主键的非主属性。

练习一

一、选择题

- 一个完整的数据模型主要包括（ ）三部分。
 - 现实世界、信息世界和机器世界
 - 表结构、记录和字段
 - 数据结构、数据操作和完整性约束
 - 数据库、表和记录
- E-R 模型属于（ ）。
 - 信息模型
 - 层次模型
 - 关系模型
 - 网状模型
- 用二维表来表示实体之间联系的模型称为（ ）。
 - 关系模型
 - 层次模型
 - 网状模型
 - 运算模型
- 关系模型（ ）。
 - 只能表示实体间一对一联系
 - 只能表示实体间一对多联系
 - 不能表示实体间多对多联系
 - 可表示实体间的任意联系方式
- 关系数据库的关系操作主要有三种，它们是（ ）操作。
 - 选择、投影、运算
 - 选择、运算、连接
 - 选择、投影、连接
 - 投影、运算、连接
- 关系模型具有某些特点，下面（ ）说法是不对的。
 - 表中不允许有重复的字段名
 - 表中每一列数据的类型必须相同
 - 表中允许有相同的记录内容
 - 表中行、列次序可以任意排列
- 数据完整性是指（ ）。
 - 实体完整性、域完整性、参照完整性
 - 字段完整性、数据范围完整性、表间关系完整性
 - 数据类型一致、数据范围一致、数据精度一致
 - 数据不能有缺陷
- 在数据库的有关概念当中，数据库（DB）、数据库系统（DBS）、数据库管理系统（DBMS）三者之间的关系是（ ）。
 - DBS 包括 DB 和 DBMS
 - DB 包括 DBS 和 DBMS
 - DBMS 包括 DB 和 DBS
 - DBS 也是 DB 或 DBMS

二、填空题

- 一个完整的数据模型包括数据结构、_____和数据完整性约束三个部分。
- 常见的结构数据模型有三种，即层次模型、网状模型和_____。
- 关键字也称为“键”，在关系表中，能够_____记录的字段或多个字段的组合称为关键字。
- 数据完整性通常包括_____、_____和参照完整性。
- 关系数据库的操作主要有选择、_____和_____三种。

6. 常见的函数依赖有完全函数依赖、_____、传递函数依赖三种。

三、设计题

1. 请使用关系规范化技术对如下“订单信息表”进行规范化处理，要求规范化后的关系模式能满足第三范式（3NF）的要求。

订单号	销售员	部门	订货量	订货日期	交货日期	客户	客户电话	货品名称	售价	成本价	供货商
8001	张明	销售部	20	2008-01-11	2008-03-21	王虹	136...	电脑（P4）	5100	4500	联想
8002	李洪	生产部	10	2009-01-01	2009-03-01	李明	159...	打印机	4300	4000	爱普新
8003	王名利	销售部	5	2009-02-01	2009-05-01	张字	133...	显示器	2300	1900	三星
8004	李洪	生产部	20	2009-04-11	2009-05-20	王虹	136...	电脑（P5）	5100	4500	联想司
8005	张明	销售部	10	2009-05-10	2009-05-25	张字	133...	显示器	2300	1900	三星

2. 根据你对“行包托运”业务的了解情况，使用 E-R 图技术为从事行包托运的企业设计一个“行包托运管理系统”的信息模型，然后将 E-R 模型转化为关系模式表示。