第 1 章　数据库基础知识

在现代社会中，各个企业或组织中都存在大量需要管理的数据，例如列车、航空的票务数据，银行中储户的账户数据等，这些数据经常会被检索、修改或删除，例如，银行储户的取款行为会涉及数据的检索和修改操作，以及数据使用过程中的安全问题。那么如何管理这些数据才能使其快捷、安全地为人们的学习、生活、工作提供服务？数据库正是基于此类需求而研发的技术。数据库是最新的数据管理技术，今天，数据库已经被应用在社会的各个领域的数据管理中。

本章首先介绍数据库的三个基本概念：数据库、数据库管理系统和数据库系统，然后讲解如何在数据库中表达现实世界中的事物。

1.1　数据库系统概述

1.1.1　数据库技术的发展

数据库技术产生于 20 世纪 60 年代后期，但在数据库技术出现之前，计算机中的数据管理经历了人工管理和文件系统两个阶段。在这两个阶段中，数据分别由应用程序和文件系统进行管理，其管理方式存在一定的缺点，以文件系统阶段为例，它存在着数据共享性和独立性差等缺点。数据共享性差会导致大量的数据冗余，浪费存储空间，而且由于数据的重复存储，还易造成数据的不一致。数据的独立性差会导致难以增加新的应用，系统扩充困难，而且当数据的结构发生变化时，需要修改应用程序以适应新的数据结构。

20 世纪 60 年代中后期，计算机管理的数据规模越来越大，应用范围也越来越广，数据量激增；在处理方式上，联机实时处理的需求也越来越多。在此背景下，文件系统管理数据的方式已不能满足应用的需求，数据库技术正是在此背景下产生的。

自数据库技术产生至今，其发展经历了 3 个阶段，即第一代的层次、网状数据库系统，第二代的关系数据库系统，以及第三代的数据库系统。数据库发展阶段的划分是以数据模型的发展为主要依据的。数据模型的发展经历了格式化数据模型（层次数据模型和网状数据模型）、关系数据模型两个阶段，并正向面向对象数据模型等非传统数据模型阶段发展。

1．第一代数据库系统

第一代数据库系统指的是以层次和网状模型为基础的层次和网状数据库系统，产生于 20 世纪 60 年代末期，它们是最早研究的数据库系统。

层次数据库是数据库系统的先驱，其代表系统为 1969 年 IBM 公司开发的 IMS 数据库系统。

1969 年，美国数据库系统语言协会 CODASYL（Conference On Data System Language）的数据库研制者提出了网状模型数据库系统规范报告，称为 DBTG（Data Base Task Group）

报告，使数据库系统开始走向规范化和标准化，这是网状数据库系统的典型代表。网状数据库是数据库概念、方法和技术的奠基者。

2．第二代数据库系统

第二代数据库系统指的是支持关系模型的关系数据库系统。

1970 年，E.F.Codd 发表了题为《A Relational Model of Data for Shared Data Banks》的论文，提出了数据库的关系模型，开始了数据库关系方法和关系数据理论的研究，为关系数据库技术奠定了理论基础。

关系数据库系统以关系代数为语言模型，以关系数据库理论为其理论基础，具有形式化基础好、数据独立性强及数据库语言非过程化等特点。

3．第三代数据库系统

与第一代、第二代数据库系统不同，第三代数据库系统没有统一的数据模型，但其数据模型具有面向对象模型的基本特征。对象关系数据库、面向对象数据库、并行数据库、空间数据库等都可以广泛称为第三代数据库系统。除了传统的数据管理服务外，第三代数据库可支持更加丰富的对象结构和规则，集数据管理、对象管理和知识管理为一体，可满足更加广泛复杂的新应用的要求。

1.1.2　数据库新技术

计算机相关技术的发展以及应用领域的变化推动着数据库技术不断向前发展，如分布式数据库、并行数据库、移动数据库和 Web 数据库等都是数据库与其他技术相结合所产生的新型数据库系统，而工程数据库、空间数据库、统计数据库和数据仓库等则是为适应特定应用领域的需求而产生的数据库新技术。

1．分布式数据库

分布式数据库系统是在集中式数据库系统的基础上发展来的，是数据库技术与网络技术结合的产物。可以对分布式数据库给出如下定义：

分布式数据库是由一组数据组成的，这组数据分布在计算机网络的不同计算机上，网络中的每个结点都具有独立处理的能力（即场地自治），可以执行局部应用。同时，每个结点也能通过网络通信子系统执行全局应用。

在该定义中，强调了分布式数据库的场地自治性和场地之间的协作性。也就是说，每个场地都是独立的数据库系统，拥有自己的数据库、自己的用户，运行自己的 DBMS，执行局部应用，具有高度的自治性。并且，各个场地之间的数据库系统又相互协作组成一个整体：即从用户的角度看，一个分布式数据库系统在逻辑上和集中式数据库系统是一样的，用户可以在任何一个场地执行全局应用。就好像那些数据是存储在同一台计算机上，由单个数据库管理系统来管理一样，用户并没有什么不同的感觉。

分布式数据库具有以下优点：

（1）更适合分布式的管理与控制；

（2）具有灵活的体系结构；

（3）系统经济，可靠性高，可用性好；

（4）局部应用的响应速度快；

（5）可扩展性好，易于集成现有系统，易于扩充。

2．Web 数据库

Web 数据库是 Web 技术与数据库技术相融合的结果，是一个以后台数据库为基础，加上一定的前台程序，通过浏览器完成数据库存储、查询等操作的系统。简单地说，一个 Web 数据库就是用户利用浏览器作为输入接口，输入所需要的数据，然后将这些数据传送给网络，网站再对这些数据进行处理，例如将数据存入数据库，或者对数据库进行查询操作等，最后网站将操作结果传回给浏览器，通过浏览器将结果反馈给用户。

与传统方式相比，利用 Web 来访问数据库，具有以下优点。

（1）利用通用的浏览器软件实现数据库客户端功能，不再需要考虑数据库客户端的设计，软件更新更加方便。

（2）数据库与浏览器完全独立，数据库结构的变更不会影响浏览器软件。因此，用户的操作不受影响。

（3）标准统一。HTML 语言是网络上的信息组织方式，是一种国际标准语言，所有的浏览器软件都遵循这个标准。

（4）具有跨平台特性。每种操作系统下都有浏览器软件可供使用。因此，设计开发的 Web 数据库应用可以在各种平台下运行，从而提高了企业软、硬件选择的自由度。

3．数据仓库

数据仓库就是面向主题的、集成的、相对稳定的、随时间不断变化（不同时间）的数据集合，用以支持经营管理中的决策制定过程。

数据仓库具有如下特征。

（1）面向主题。

数据仓库中的数据面向主题，与传统数据库面向应用相对应。主题是一个在较高层次上将数据归类的标准，每一个主题对应一个宏观的分析领域，数据仓库中的数据是面向主题进行组织的。面向主题的数据组织方式，就是在较高层次上对分析对象的数据的一个完整、一致的描述，能完整、统一地刻画各个分析对象所涉及的各项数据及数据间的联系。

（2）集成化特性。

数据仓库中的数据是从原有分散的数据库中抽取出来的，由于数据仓库的每一主题所对应的源数据在原有分散的数据库中可能有重复或不一致的地方，加上综合数据不能从原有数据库中直接得到。因此数据在进入数据仓库之前必须要经过统一和综合形成集成化的数据。这是建立数据仓库的关键步骤，不但要统一原始数据中的矛盾之处，还要将原始数据结构做一个从面向应用向面向主题的转变。

（3）稳定性。

数据仓库的稳定性是指数据仓库反映的是历史数据，而不是日常事务处理产生的数据，数据经加工和集成进入数据仓库后是极少或根本不修改的。

（4）随时间不断变化。

数据仓库中数据的不可更新性是针对应用来说的，即用户进行分析处理时是不进行数据更新操作的。但并不是说，在数据仓库的整个生存周期中数据库集合是不变的。

数据仓库会随时间的变化不断增加新的数据内容，以及删除旧的数据内容。而且，数据仓库中包含大量的综合数据大多与时间有关，这些数据会随着时间的变化不断地重新进行综合，这些数据的码键都包含时间项，以标明数据的历史时期。所以，数据仓库中的数据是随时间不断变化的。

1.1.3 数据库相关基本概念

为了更好地使用数据库，首先需要了解数据库、数据库管理系统、数据库系统等基本概念。

1．数据

信息是现实世界中人们对客观事物状态和特征的描述。数据（Data）是承载信息的符号记录，它是数字、字母、文字、图像、声音、视频等信息的描述形式，常常经过数字化处理后存入计算机来反映或描述事物的特性。

2．数据库

简单地说，数据库（DataBase，DB）就是存放数据的仓库。在现代社会中，数据的规模越来越大，将数据存储在数据库中，可以更加方便、快捷，并且充分利用这些数据。

严格地说，数据库是指长期存储在计算机内、有组织的、可共享的大量数据的集合。数据库中的数据按照一定的数据模型进行组织、描述和存储，并具有较小的冗余度、较高的数据独立性，且可由各种用户共享。

3．数据库管理系统

了解了数据库的基本概念之后，接下来的问题是数据库如何存储在计算机中，如何才能高效地检索和维护数据库中的数据。解决这些问题需要的就是数据库管理系统。

数据库管理系统（Database Management System，DBMS）是一个系统软件，是提供建立、管理、维护和控制数据库功能的一组计算机软件。数据库管理系统的目标是使用户能够科学地组织和存储数据，能够从数据库中高效地获得需要的数据，方便地处理数据。

数据库管理系统主要提供以下几个方面的功能：

（1）数据定义功能。

数据库管理系统提供数据定义语言，用户通过它可方便地对数据库中的数据对象进行定义。

（2）数据组织、存储和管理。

数据库管理系统会分类组织、存储和管理各种数据，并确定以何种文件结构和存取方式在存储级上组织数据，其目的是提高存储空间的利用率和数据的存取效率。

（3）数据操纵功能。

数据库管理系统通过提供数据操纵语言实现对数据的增、删、改、查询、统计等数据

操纵功能。

（4）数据库的建立和维护功能。

数据库管理系统包括数据库初始数据输入、转储、恢复、重组以及数据库结构的修改和扩充等功能。

（5）数据库的运行管理。

数据库的运行管理功能是数据库管理系统的核心功能，它对数据库的建立、运行和维护进行统一管理，保证数据的安全性、完整性、并发性和故障恢复。

4．数据库系统

（1）数据库系统的组成。

仅有数据库管理系统，是不能完成数据库的建立、使用和维护等工作的，一个完整的数据库系统还应包括除数据库管理系统之外的元素。一般来说，数据库系统（DataBase System，DBS）是指带有数据库并利用数据库技术进行数据管理的计算机系统，包括以下 4 部分：

- 数据库：数据库系统的数据源。
- 硬件：支持系统运行的计算机硬件设备。包括 CPU、内存、外存及其他外部设备。
- 软件：包括操作系统、数据库管理系统、应用开发工具和应用系统。
- 人员：数据库系统中的主要人员有数据库管理员、系统分析员和数据库设计人员、应用程序开发人员和最终用户。

（2）数据库系统的特点。

与人工管理和文件系统相比，数据库系统主要有以下 4 个特点：

- 数据结构化。在数据库系统中，数据是面向整体的，不但数据内部组织有一定的结构，而且数据之间的联系也按一定的结构描述出来，所以数据整体结构化。
- 数据共享性高，冗余度低，易扩充。数据库系统是面向整体的，因此数据可以被多个用户共享使用，大大减少了冗余度。而且可以很容易地增加新的功能，适应用户新的要求。
- 数据独立性高。数据库系统的体系结构包括三级模式和两级映射，保证了程序与数据库中的逻辑结构和物理结构有高度的独立性。
- 数据由数据库管理系统统一管理和控制。数据库管理系统在数据库建立、运用和维护时对数据库进行统一控制，以保证数据的完整性、安全性，并在多用户同时使用数据库时进行并发控制，在发生故障后对系统进行恢复。

5．数据库应用系统

数据库应用系统（DataBase Application System）就是利用数据库技术管理数据的系统，它是在数据库管理系统支持下建立的计算机应用系统。数据库应用系统包括：应用系统、应用开发工具软件、数据库管理系统、操作系统、硬件、数据库管理员、应用界面。通常，这 7 个部分以一定的逻辑层次结构方式组成一个有机的整体。概括地说，数据库应用系统就是利用数据库技术，面向某个特定应用开发的应用软件及相关，例如财务管理系统、人事管理系统、图书管理系统、教学管理系统等。

1.2 数据模型

数据库中的数据来源于现实世界，那么，在实现数据库系统时，需要考虑的问题是：如何描述现实世界中的事物，才能在数据库中清晰、准确地表达现实世界中的事物以及事物间的联系。这就需要使用数据模型来解决这一问题。

数据模型是数据特征的抽象，它是对数据库如何组织的一种模型化表示。计算机不可能直接处理现实世界中的具体事物，人们必须把具体事物转换成计算机能够处理的数据，因此人们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。无论处理任何数据，都要先对数据建立模型，然后在此基础上进行处理。

数据模型应满足 3 方面要求：一是能比较真实地模拟现实世界；二是容易为人所理解；三是便于在计算机上实现。

根据模型应用的不同目的，可以将模型分为两类：一类模型是概念模型，也称信息模型，它按用户的观点来对数据和信息建模，主要用于数据库设计。概念模型不依赖于具体计算机系统，也不是某一种数据库管理系统支持的模型。另一类模型是逻辑模型，它按计算机系统的观点对数据建模，主要用于数据库管理系统的实现。数据模型描述数据的结构、定义在其上的操作以及约束条件。它具有数据结构、数据操作和数据的完整性约束三要素。

1.2.1 概念模型

在实现数据库系统的时候，需要先把现实世界中的事物抽象成概念模型，然后再把概念模型转换为计算机上某一种数据库管理系统支持的数据模型。

概念模型用于信息世界的建模，是现实世界到信息世界的第一层抽象，是现实世界到机器世界的一个中间层次。概念模型应该简单、清晰、易于用户理解，还应该具有较强的语义表达能力，能够方便、直接地表达应用中的各种语义。

1．信息世界中的基本概念

在使用概念模型对现实世界进行抽象之前，首先需要了解以下与概念模型相关的主要概念：

（1）实体。

客观存在并可相互区别的事物称为实体。例如，一门课程、一个学生等。

（2）属性。

实体所具有的某一特性称为属性。例如，学生的学号、姓名。

（3）关键字。

唯一标识实体的属性集称为关键字。例如，学号是学生实体的关键字。

（4）实体型。

具有相同属性的实体必然具有共同的特征和性质。用实体名及其属性名集合来抽象和刻画同类实体，称为实体型。例如，学生（学号，姓名，性别，出生年份，系，入学日期）就是一个实体型。

（5）实体集。

同型实体的集合称为实体集。例如，全体学生就是一个实体集。

（6）联系。

在现实世界中，事物内部以及事物之间是有联系的，这些联系在信息世界中反映为实体（型）内部的联系和实体（型）之间的联系。

两个实体型之间的联系可以分为三类：

（1）一对一联系（1:1）。

如果对于实体集 A 中的每一个实体，实体集 B 中至多有一个（也可以没有）实体与之联系，反之亦然，则称实体集 A 与 B 具有一对一联系，记为 1:1。

例如，一个班级只有一个正班长，而一个班长也只在一个班中任职，如图 1.1（a）所示。

（2）一对多联系（1:*n*）。

如果对于实体集 A 中的每一个实体，实体集 B 中有 *n* 个实体（*n*≥0）与之联系，反之，对于实体 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 B 有一对多联系，记为 1:*n*。

例如，一个班级中可以有若干名学生，而每个学生只在一个班级中学习，如图 1.1（b）所示。

（3）多对多联系（*m*:*n*）。

如果对于实体集 A 中的每一个实体，实体集 B 中有 *n* 个实体（*n*≥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*≥0）与之联系，则称实体集 A 与 B 具有多对多联系，记为 *m*:*n*。

例如，一个学生可以选修多门课程，而一门课程可以被多个学生选修，学生和课程之间就是多对多的联系，如图 1.1（c）所示。

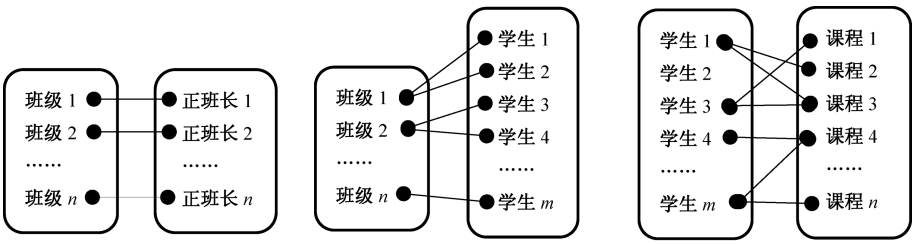

图 1.1　两个实体型之间的联系

2．概念模型的表示方法

在对现实世界进行建模之后，需要将建立的概念模型表达出来。表示概念模型的方法很多，其中最常用的是实体-联系方法（Entity-Relationship Approach），该方法使用 E-R 图来表示概念模型。

E-R 图提供了表示实体型、属性和联系的方法：

（1）实体型。

使用矩形表示实体型，矩形内写明实体名。

（2）属性。

使用椭圆表示属性，并用无向边将其与相应的实体型连接起来。

例如：学生实体具有学号、姓名、出生日期、性别、入学日期等属性，用 E-R 图表示学生实体如图 1.2 所示。

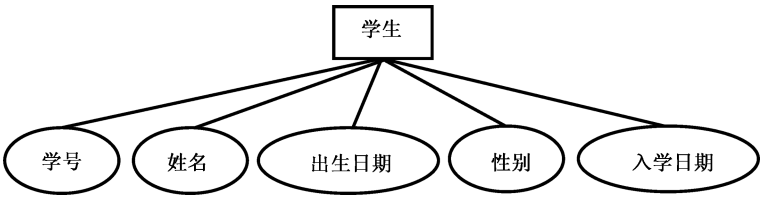

图 1.2　学生实体及其属性

（3）联系。

使用菱形表示，菱形内写明联系名，并用无向边分别与有关实体型连接起来，并在无向边旁标注联系的类型（1:1，1:*n*，*m*:*n*）。

例如：学生实体和课程实体之间存在 *m*:*n* 联系，且该联系具有一个“成绩”属性，如图 1.3 所示。

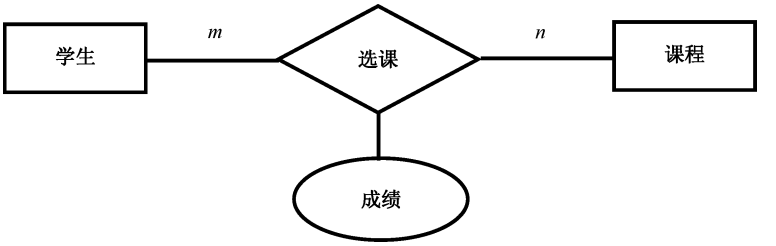

图 1.3　实体联系

1.2.2　数据模型的三要素

数据模型的组成要素有 3 个，分别是数据结构、数据操作和数据完整性约束条件。

（1）数据结构。

数据结构是对系统静态特征的描述。主要描述数据类型、内容、性质以及数据间联系的有关信息。数据结构是数据模型的基础，在数据库系统中，通常按照数据结构的类型来命名数据模型，例如，数据结构为层次、网状或关系结构的数据模型分别被命名为层次模型、网状模型和关系模型。

（2）数据操作。

数据操作描述的是系统的动态特征，主要描述在相应数据结构上的操作类型与操作方式。数据操作主要有数据检索和更新（即增、删、改）两大类操作。

（3）数据完整性约束条件。

数据完整性约束条件描述的是系统的约束条件，主要描述数据结构内数据间的语义限制、制约与依存关系以及数据动态变化的规则以保证数据的正确、有效与相容。

1.3 关系模型

在数据库的发展史上，主要的逻辑数据模型包括层次模型、网状模型和关系模型等三种模型。目前，主流的数据库管理系统大多是基于关系模型的。基于关系模型的数据库管理系统称为关系型数据库管理系统。接下来，首先通过关系模型的三要素：关系数据结构、关系操作、关系完整性约束条件来了解关系模型，然后介绍 E-R 图向关系模型的转换，以及关系模型的规范化。

1.3.1 关系数据结构

在关系模型中，无论实体还是实体之间的联系都由单一的数据结构即关系（表）来表示。

1．关系模型的基本术语

关系：关系模型中一个关系就是一个二维表，每个关系有一个关系名，如图 1.4 所示的表格即为一个关系，此关系名为“学生信息表”。

元组：表中的一行即为一个元组，如“学生信息表”的一个元组（001，李月，1994-1-5，女，2012.9）。

学号	姓名	出生日期	性别	入学日期
001	李月	1994-1-5	女	2012.9
002	王明	1993-12-3	男	2012.9
003	孙杰	1994-1-6	男	2012.9
…	…	…	…	…

图 1.4 “学生信息表”关系数据结构

属性：表中的一列即为一个属性，给每个属性起一个名字即为属性名，如“学号”、“姓名”等属性。

域：属性的取值范围，如性别的域是（男，女），百分制成绩的域是 0~100。

关键字：属性或属性的集合，其值能唯一地标识一个元组。例如，“学生信息表”的“学号”属性在该关系中具有唯一性，可以作为该关系的关键字。

外关键字：若一个关系 R 中的属性（或属性组）F 不是其关键字，却与另一个关系 S 的主关键字 Ks 相对应，则 F 称为是 R 关系的外关键字。例如，在“教学管理系统”中，“班级”字段在“学生信息表”中不是主关键字，但是“班级信息表”的“班级编号”是主关键字，而且此“班级编号”与“学生信息表”的“班级”字段相对应，则称“班级”是“学生信息表”的外关键字。

关系模式：关系名及关系的属性集合构成关系模式，一个关系模式对应一个关系的结构。关系模式的格式为：关系名（属性 1，属性 2，…，属性 *n*）。例如，学生信息表的关系模式为：学生信息表（学号，姓名，性别，密码，出生日期，民族，籍贯，政治面貌，入学日期，班级，照片，备注）。

2．关系的基本性质

- 关系中的每一列是同一类型的数据，来自同一个域。
- 关系中的每一列称为一个属性，不同的属性要给予不同的属性名。
- 列的顺序无所谓，即列的次序可以任意交换。
- 关系中的每一行称为一个元组，任意两个元组不能完全相同。
- 行的顺序无所谓，即行的次序可以任意交换。

1.3.2 关系操作

关系的基本运算可分为两类：传统的集合运算（并、差、交等）和专门的关系运算（选择、投影、连接）。

1．传统的集合运算

传统的集合运算包括并、交、差和笛卡儿乘积等运算。注意，进行并、差和交运算的两个关系必须具有相同的关系模式，即元组有相同结构。学生信息表 1 和学生信息表 2 就是两个具有相同结构的关系，其结构和关系中的元组如表 1.1 和表 1.2 所示。下面将以这两个表为例介绍关系运算。

表 1.1　学生信息表 1

学号	姓名	出生日期
001	李月	1994-1-5
002	王明	1993-12-3
003	孙杰	1994-1-6

表 1.2　学生信息表 2

学号	姓名	出生日期
001	李月	1994-1-5
004	张力	1993-2-12
005	刘丽	1994-3-20

（1）并。

关系 R 与 S 的“并”是由属于 R 或 S 的元组组成的集合，并运算由符号“∪”表示，关系 R 和关系 S 的并可表示为 R∪S。

例如：学生信息表 1∪关系学生信息表 2 的结果如表 1.3 所示。

（2）交。

关系 R 和 S 的“交”是由既属于 R 又属于 S 的元组组成的集合。交运算的结果是 R 和 S 的共同元组。

交运算由符号“∩”表示，关系 R 和关系 S 的交可表示为 R∩S，例如：学生信息表 1 和学生信息表 2 的交可表示为“学生信息表 1∩学生信息表 2”，其结果如表 1.4 所示。

表 1.3　学生信息表 1∪关系学生信息表 2

学号	姓名	出生日期
001	李月	1994-1-5
002	王明	1993-12-3
003	孙杰	1994-1-6
004	张力	1993-2-12
005	刘丽	1994-3-20

表 1.4　学生信息表 1∩学生信息表 2

学号	姓名	出生日期
001	李月	1994-1-5

（3）差。

关系 R 与 S 的“差”是由属于 R 但不属于 S 的元组组成的集合，即差运算的结果是从 R 中去掉 R 和 S 共同包含的元组。

差运算由符号“－”表示，，关系 R 和关系 S 的差可表示为 R－S，例如：学生信息表 1 和学生信息表 2 的差可表示为“学生信息表 1－学生信息表 2”，其结果如表 1.5 所示。

（4）笛卡儿乘积。

关系 R 和 S 的笛卡儿乘积是由 R 中的每一个元组与 S 中的任意元组组合而成的元组组成的集合，该集合中元组的个数为 $m \times n$，其中 m 为 R 的元组数，n 为 S 的元组数。

笛卡儿乘积用符号“×”表示，关系 R 和关系 T 的笛卡儿乘积可表示为 R×T，假设成绩信息表的结构和关系中的元组如表 1.6 所示，则学生信息表 1 和成绩信息表的笛卡儿乘积可表示为“学生信息表 1×成绩信息表”，其结果如表 1.7 所示。

表 1.5　学生信息表 1－学生信息表 2

学号	姓名	出生日期
002	王明	1993-12-3
003	孙杰	1994-1-6

表 1.6　成绩信息表

学号	课程编号	成绩
001	C001	89
002	C002	85

表 1.7　学生信息表 1×成绩信息表

学生信息表 1.学号	姓名	出生日期	成绩信息表.学号	课程编号	成绩
001	李月	1994-1-5	001	C001	89
001	李月	1994-1-5	002	C002	85
002	王明	1993-12-3	001	C001	89
002	王明	1993-12-3	002	C002	85
003	孙杰	1994-1-6	001	C001	89
003	孙杰	1994-1-6	002	C002	85

请同学们观察表 1.7 每行的数据，说一说哪些数据是有意义的。

2．专门的关系运算

（1）选择。

从关系中查找出满足给定条件的元组的操作称为选择。

选择的条件以逻辑表达式给出，选择运算的结果是由逻辑表达式的值为真的元组组成的集合。

例如，从学生信息表 1 中选择出所有姓李的学生的信息，其结果如表 1.8 所示。

表 1.8　选择运算

学号	姓名	出生日期
001	李月	1994-1-5

（2）投影。

从关系中选择出若干属性的操作称为投影。

例如，学生信息表 1 中选择出学生的学号、姓名，其结果如表 1.9 所示。

表 1.9　投影运算

学号	姓名
001	李月
002	王明
003	孙杰

（3）连接。

连接运算是从两个关系的笛卡儿乘积中选择出满足指定条件的元组。

例如，学生信息表 1 和成绩信息表按照“学生信息表 1.学号=成绩信息表.学号”进行连接运算，其结果如表 1.10 所示。

表 1.10　连接运算

学生信息表 1.学号	姓名	出生日期	成绩信息表.学号	课程编号	成绩
001	李月	1994-1-5	001	C001	89
002	王明	1993-12-3	002	C002	85

从连接运算可以看出，当我们需要的数据分别存储在不同的表时，可以使用连接运算将不同表中的元组连接在一起。例如，如果想知道学生“李月”的“成绩”，“李月”这个姓名存储在“学生信息表 1”中，而属性“成绩”存储在“成绩信息表”中，所以需要将“学生信息表 1”和“成绩信息表”连接起来，才能得到所需要的数据即李月的成绩。

在该例中，两个关系的连接条件是“学生信息表 1..学号=成绩信息表学号”，使用的关系运算符为“=”，这样的连接称为等值连接。

在连接运算中，按照字段值对应相等为条件进行的连接操作为等值连接。连接结果中去掉重复值的等值连接叫自然连接，自然连接是最常用的连接运算。

1.3.3　关系的完整性

关系模型的完整性规则是对关系的一种约束条件。关系模型存在三类完整性约束：实体完整性、参照完整性和用户定义的完整性。

1．实体完整性

实体完整性是指关系的主属性不能取空值，即主属性不能是“不知道”或“不存在”的值。

例如在关系学生信息表（学号、姓名、出生年月、班级）中，“学号”为该关系的主属性，则“学号”不能取空值。

根据实体完整性的要求，如果关系的主关键字由若干属性组成，则所有这些主属性都不能取空值。例如学生选课关系“选课（学号，课程编号，成绩）”中，“学号、课程编号”为主属性，则“学号”和“课程编号”都不能取空值。

2．参照完整性

参照完整性规则定义了外关键字与主关键字之间的引用规则。下面首先给出外关键字

的定义。

外关键字：如果关系 R 中的一个或一组属性 F 不是 R 的关键字，但却与关系 S 的主关键字相对应，则 F 称为 R 的外关键字。

参照完整性的含义为：

若属性（或属性组）F 是基本关系 R 的外关键字，它与基本关系 S 的主关键字 Ks 相对应（基本关系 R 和 S 不一定是不同的关系），则对于 R 中每个元组在 F 上的值必须为：

- 或者取空值（F 的每个属性值均为空值）；
- 或者等于 S 中某个元组的主关键字值。

例如，“教学管理系统”中的两个表：学生信息表、班级信息表的关系模式如下：

学生信息表（学号，姓名，性别，密码，…，入学日期，班级，照片，备注）

班级信息表（班级编号、班级名称，学生数，所属学院）

学生信息表的主关键字为“学号”，班级信息表的主关键字为“班级编号”，“班级”是关系“学生信息表”的外关键字，它参照了班级信息表中的“班级编号”属性，如图 1.5 所示。这时，学生信息表中的“班级”属性的取值只有以下两种情况。

（1）空值：表示还没有给学生分配班级；

（2）取班级信息表中的“班级编号”属性中存在的值：表示该学生被分配在某一个存在的班级中。

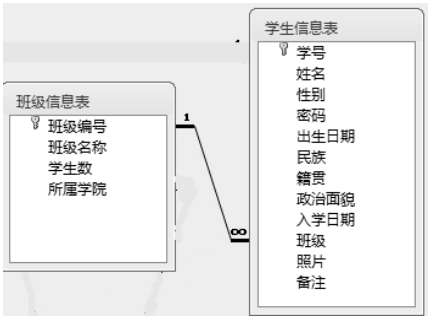

图 1.5　关系的参照图

3．用户定义的完整性

不同的关系数据库系统根据其应用环境的不同，往往还需要一些特殊的约束条件，用户定义的完整性就是针对某一具体关系数据库的约束条件。例如，定义“成绩”字段的取值范围为 0～100。

1.4　数据库设计

数据库设计是指针对给定的应用环境，构造优化的数据库逻辑模式和物理结构，并据此建立数据库及其应用系统，使其能够有效地存储和管理数据，满足用户的应用需求。数据库的设计和开发是一项庞大的工程，需要用到信息资源管理、软件开发工具、数据库理论等基础知识。

1.4.1 数据库设计的步骤

数据库应用系统以数据库为核心和基础，数据库设计要与整个数据库应用系统的设计开发结合起来进行，只有设计出高质量的数据库，才能开发出高质量的数据库应用系统，也只有着眼于整个数据库应用系统的功能要求，才能设计出高质量的数据库。

数据库设计包括需求分析、概念结构设计、逻辑结构设计、物理结构设计、数据库实施、数据库运行和维护 6 个阶段。

（1）需求分析。

需求分析的任务是通过详细调查现实世界要处理的对象（组织、部门、企业等），充分了解原系统（手工系统或计算机系统）工作概况，明确用户的各种需求，然后在此基础上确定新系统的功能。新系统必须充分考虑今后可能的扩充和改变，不能仅按当前营业需求来设计数据库。这里的重点是对建立数据库的必要性及可行性进行分析和研究，确定数据库在整个数据库应用系统中的地位以及各个数据库之间的关系。

（2）概念结构设计。

概念结构设计是整个数据库设计的关键，它通过对需求分析阶段得到的用户需求进行综合、归纳和抽象，形成一个独立于具体 DBMS 的概念模型。

（3）逻辑结构设计。

逻辑结构设计就是把概念结构设计阶段的 E-R 图转换成与具体的数据库管理系统产品所支持的数据模型相一致的逻辑结构。逻辑结构设计包括两个步骤：将 E-R 图转换为关系模型和对关系模型进行优化。

（4）物理结构设计。

数据库在实际的物理设备上的存储结构和存取方法称为数据库的物理结构。对于设计好的逻辑模型选择一个最符合应用要求的物理结构就是数据库的物理结构设计，物理结构设计依赖于给定的硬件环境和数据库产品。

（5）数据库实施。

数据库实施阶段的工作就是根据逻辑设计和物理设计的结果，在选用的 DBMS 上建立起数据库，主要包括建立数据库的结构、载入实验数据并测试应用程序、载入全部实际数据并试运行应用程序等几项工作。

（6）数据库运行和维护。

数据库经过试运行就可以投入实际运行了。但是，由于应用环境在不断变化，对数据库设计进行评价、调整、修改等维护工作是一个长期的任务，也是设计工作的继续和提高。

1.4.2 概念结构设计

概念结构设计是指将需求分析阶段得到的用户需求抽象为信息结构即概念模型的过程。它是整个数据库设计的关键。

概念结构有以下一些特点：

- 能真实、充分地反映现实世界。
- 易于理解，因而可以以此为基础和不熟悉数据库专业知识的用户交换意见。

- 当应用环境和用户需求发生变化时，很容易实现对概念结构的修改和完善。
- 易于转换成关系、层次、网状等各种数据模型。

概念结构从现实世界抽象而来，又是各种数据模型的共同基础，实际上是现实世界与逻辑结构（机器世界）之间的一个过渡。

描述概念模型常用的工具是 E-R 模型。下面将使用 E-R 模型来描述生成的概念结构。

概念结构是对现实世界的一种抽象，也就是抽取现实世界中人、物、事等的共同特征，并把这些特征用各种概念精确地加以描述。在建立概念结构的过程中，可以先根据对象的特征和行为分类对象，具有共同特征和行为的对象作为一类，每一类都是概念模型中的一个实体型。例如，在学校的教学管理中，学生都具有相同的特征，如学号、姓名、班级等，也具有相同的行为，如选修课程；教师具有相同的特征，如教师编号、姓名、职称等，同时也具有相同的行为，如教学。根据分类的方法，可以把教学管理中的人和事物分为以下几个实体型：学生、教师、班级、课程、学院，它们具有的属性如下：

学生（学号，姓名，登录密码，出生日期，性别，入学日期，…）

教师（教师编号，姓名，登录密码，职称，教学网站）

班级（班级编号，专业，学生数）

课程（课程编号，课程名称，学分，课程类别，学时，课程简介）

学院（学院编号，学院名称，院办电话）

其中，加下画线的属性是实体型的关键字属性。

图 1.6 所示是这些实体型的 E-R 图表示。

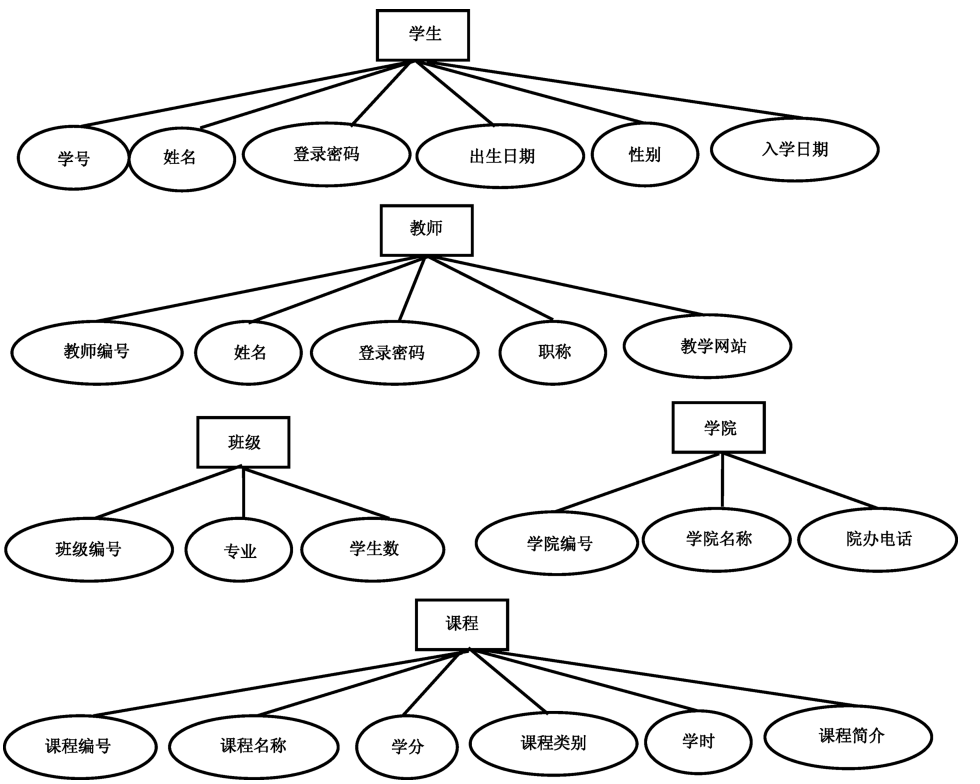

图 1.6　教学管理系统中的实体型及其属性

接下来，需要考虑实体之间的联系：

（1）班级与学生之间是一对多联系；

（2）学院与班级之间是一对多联系；

（3）学生与课程之间是多对多联系；

（4）教师与课程、班级三者之间是多对多联系；

（5）学院与教师之间存在两种联系：一对多联系和一对一联系。一对多联系指的是一个学院中可以有多位教师，而一位教师只能在一个学院中任职；一对一联系指的是一个学院只能有一个正院长，而一个正院长只能管理一个学院（正院长也是教师中的一员）。

用 E-R 图表示这些实体型之间的联系如图 1.7 所示。

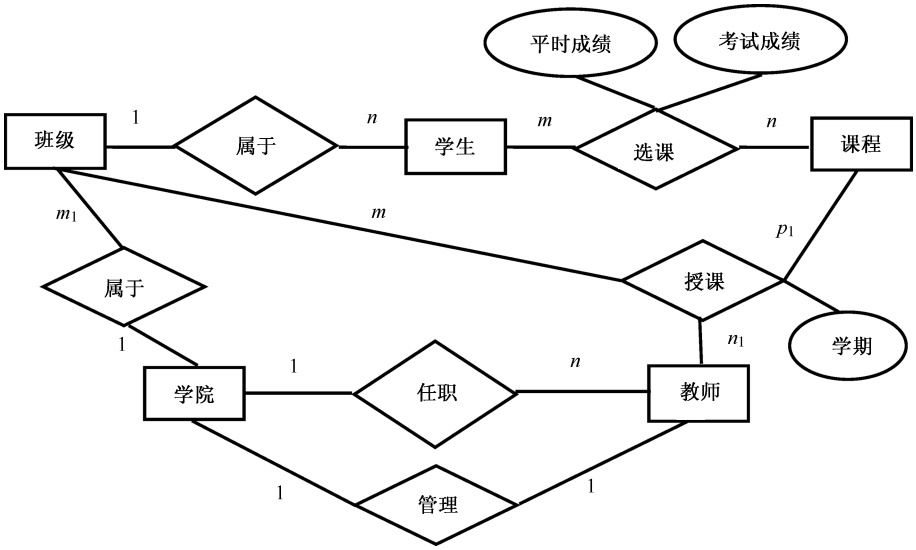

图 1.7　教学管理系统中的实体联系

1.4.3　概念模型向关系模型的转换

使用 E-R 图可以清晰地对现实世界中的事物进行分类，并清晰地描述事物之间的联系，便于人们的理解以及更完整地了解现实世界中的事物。但 E-R 图并不能被计算机所识别，因此，还需要将 E-R 图转换为数据库管理系统所支持的逻辑数据模型。

E-R 图向关系模型的转换实际上就是将 E-R 图中的实体型、实体的属性和实体型之间的联系转换为关系模式，这种转换一般遵循以下原则。

（1）一个实体型转换为一个关系模型，实体的属性就是关系的属性，实体的关键字就是关系的关键字。

（2）实体型之间联系的转换。

① 1:1 联系。

一个 1:1 联系可以转换为一个独立的关系模式，也可以与任意一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关键字以及联系本身的属性都应转换为关系的属性。如果与某一端实体对应的关系模式合并，则需要在该关系模式的属性中加入另一个关系模式的关键字和联系本身的属性。例如，学院实体和教师实体

之间的管理联系是一个 1:1 联系，可以把它与学院关系模式合并，即在学院关系中增加一个名称为“院长”的属性，则合并后的学院关系模式为：

学院（学院编号，学院名称，院长）

② 1:*m* 联系。

一个 1:*m* 联系可以转换为一个独立的关系模式，也可以与 *m* 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关键字以及联系本身的属性都应转换为关系的属性。如果与 *m* 端实体对应的关系模式合并，则需要在该关系模式的属性中加入 1 端实体对应的关系模式的关键字和联系本身的属性。例如，学生实体与班级实体之间的联系是 1:*m* 联系，可将其与学生关系模式合并，即在学生关系模式中增加一个名称为“班级”的属性，则合并后的学生关系模式为：

学生（学号，姓名，登录密码，出生年月，性别，入学日期，班级）

③ *m*:*n* 联系。

一个 *m*:*n* 联系转换为一个关系模式。与该联系相连的各实体的关键字以及联系本身的属性都转换为关系的属性。例如，学生实体与课程实体之间的联系为 *m*:*n* 联系，可将其转换为一个新的关系模式，在该关系模式中包含学生实体和课程实体的关键字属性以及该联系本身所具有的属性，则转换后的关系模式为：

学生选课（学号，课程编号，平时成绩，考试成绩）

按照上述转换原则对教学管理系统的 E-R 图进行转换之后，得到以下关系模式：

学生（学号，姓名，登录密码，出生日期，性别，入学日期，班级，…）

教师（教师编号，姓名，登录密码，职称，教学网站，所属学院）

班级（班级编号，班级名称，学生数，所属学院）

课程（课程编号，课程名称，学分，课程类别，学时，课程简介）

学院（学院编号，学院名称，院办电话，正院长）

学生选课（学号，课程编号，平时成绩，考试成绩）

教师授课（教师编号，班级编号，课程编号，学期）

1.4.4 关系模型的规范化

为了确保关系结构设计合理，通常要对关系进行规范化设计。通过规范化设计，可以消除关系中存在的冗余。对于关系来说，存在着多种不同的规范化形式。从规范化的宽松到严格，分别为第一范式、第二范式、第三范式等。

1．第一范式

一个满足第一规范化形式的关系中的每一个属性（字段）都是不可分的数据项。第一规范化形式简称为一范式或 1NF。1NF 是关系数据库应具备的最起码的条件，如果数据库设计不能满足第一范式，就不能称为关系型数据库。

例如，表 1.11 的“学生信息表”中“课程成绩”字段是一个可以拆分的字段项，因此该表不满足第一范式的要求。

为了使其符合第一范式，成为关系数据库中的数据表，必须进行数据表的规范化处理。方法是处理表头，使其成为只有一行表头的数据表。修改后的表如表 1.12 所示，是一个满

足 1NF 的表。

表 1.11　学生信息表

学号	姓名	性别	出生日期	班级	班级人数	课程成绩		
						课程编号	课程名称	成绩
20120101	李月	女	1994-1-5	201201	40	C001	大学计算机基础	82
20120101	李月	女	1994-1-5	201201	40	C002	数据结构	85
20120102	王明	男	1993-12-3	201201	40	C001	大学计算机基础	78
20120301	孙杰	男	1994-1-6	201203	50	C001	大学计算机基础	92
20120301	孙杰	男	1994-1-6	201203	50	C015	古代汉语	86
20120301	李强	男	1994-6-3	201203	50	C001	大学计算机基础	70

表 1.12　学生信息表

学号	姓名	性别	出生日期	班级	班级人数	课程编号	课程名称	成绩
20120101	李月	女	1994-1-5	201201	40	C001	大学计算机基础	82
20120101	李月	女	1994-1-5	201201	40	C002	数据结构	85
20120102	王明	男	1993-12-3	201201	40	C001	大学计算机基础	78
20120301	孙杰	男	1994-1-6	201203	50	C001	大学计算机基础	92
20120301	孙杰	男	1994-1-6	201203	50	C015	古代汉语	86
20120301	李强	男	1994-6-3	201203	50	C001	大学计算机基础	70

2．第二范式

如果在一个满足 1NF 的关系中，所有非关键字属性都完全依赖于关键字，则称这个关系满足第二规范化形式，简称二范式或 2NF。

例如：在表 1.12 的学生信息表中，当给定“学号”和“课程编号”之后，可唯一确定一个记录，因此其关键字是“学号”和“课程编号”。但是，在表 1.12 中，非关键字属性“姓名”、“性别”、“出生日期”、“班级”和“班级人数”只依赖于“学号”，与“课程编号”无关，而非关键字属性“课程名称”仅依赖于“课程编号”，与“学号”无关，因此，在表 1.12 的学生信息表中，存在某些非关键字属性不完全依赖于关键字的情况，所以表 1.12 所示的学生信息表不满足第二范式的要求。

在数据库应用系统中如果存在不满足 2NF 的数据表，则将导致数据插入或删除异常，所以需要修改数据表，使其满足 2NF 的要求。修改方法一般是对数据表进行拆分，例如，针对表 1.12 所示的学生信息表，可以将存在依赖关系的属性单独存放在一个数据表里，所以，表 1.12 可拆分为表 1.13～表 1.15。

表 1.13　学生信息表

学号	姓名	性别	出生日期	班级	班级人数
20120101	李月	女	1994-1-5	201201	40
20120102	王明	男	1993-12-3	201201	40
20120301	孙杰	男	1994-1-6	201203	50
20120301	李强	男	1994-6-3	201203	50

表 1.14　课程信息表

课程编号	课程名称
C001	大学计算机基础
C002	数据结构
C015	古代汉语

表 1.15　学生选课表

学号	课程编号	成绩
20120101	C001	82
20120101	C002	85
20120102	C001	78
20120301	C001	92
20120301	C015	86
20120301	C001	70

3．第三范式

对于那些满足 2NF 的关系，且其非关键字属性之间不存函数依赖（即：不存在一个非关键字属性，可以确定另外一些非关键字属性），则称这个关系满足第三规范化形式，简称三范式或 3NF。

一个满足 3NF 的数据库将有效地减少数据冗余。例如：在表 1.13 所示的学生信息表中，非关键字属性“班级”可以确定非关键字属性“班级人数”的值，所以非关键字之间存在函数依赖关系，表 1.13 不满足 3NF 范式。

为了使表 1.13 满足 3NF 范式，可以将它拆分成“学生信息表”和“班级信息表”两个表，每个表对应一个对象，如表 1.16 和表 1.17 所示。

表 1.16　学生信息表

学号	姓名	性别	出生日期	班级
20120101	李月	女	1994-1-5	201201
20120102	王明	男	1993-12-3	201201
20120301	孙杰	男	1994-1-6	201203
20120301	李强	男	1994-6-3	201203

表 1.17　班级信息表

班级编号	班级人数
201201	40
201203	50

在设计表时，应该保证数据库中的所有表都能满足 2NF，并应力求绝大多数表满足 3NF。首先保证单层表头，使之成为 1NF 数据表；接着分解数据表并设定关键字，使之成为 2NF 数据表；如果包含冗余，则要继续拆分数据表以消除对非关键字段之间的函数依赖，使之成为 3NF 数据表。

一般来说，如果设计 E-R 图阶段正确表达了实体及实体间的联系，那么从 E-R 图转换而来的表会满足 3NF 的要求。而且，在设计表时，如果把握住一个表中只存放关于一个主题的信息的原则（即不要把多种不同的信息混杂在一起，如表 1.13 中就存放学生和班级两个主题的信息），也可以尽可能地避免出现不满足 3NF 的情况。

本章小结

本章首先介绍了数据库技术的发展历史、基本概念，然后介绍了概念模型和数据模型，并从数据模型三要素的角度介绍了目前广泛使用的关系模型。最后介绍了数据库设计的基本步骤，并详细介绍了概念结构的设计和概念模型向关系模型的转换，以及关系模型的规范化。通过这些基本概念的介绍，使读者对使用数据库所需要的基础知识有了一个较为清晰的认识，为以后章节的学习奠定一个良好的基础。

习题

一、选择题

1．一个教师可讲授多门课程，一门课程可由多个教师讲授，则实体教师和课程间的联系是（　　）。

A．1:1 联系　　B．1:*m* 联系　　C．*m*:1 联系　　D．*m*:*n* 联系

2．把实体-联系模型转换为关系模型时，实体之间多对多联系在模型中通过（　　）。

A．建立新的属性来实现　　B．建立新的关键字来实现

C．建立新的关系来实现　　D．建立新的实体来实现

3．对关系 S 和关系 R 进行集合运算，结果中既包含 S 中元组也包含 R 中元组，这种集合运算称为（　　）。

A．并运算　　B．交运算　　C．差运算　　D．积运算

4．在下列关系运算中，不改变关系表中的属性个数但能减少元组个数的是（　　）。

A．并　　B．选择　　C．投影　　D．笛卡儿乘积

5．关系型数据库中所谓的“关系”是指（　　）。

A．各个记录中的数据彼此间有一定的关联　　B．数据模型符合满足一定条件的二维表格式

C．某两个数据库文件之间有一定的关系　　D．表中的两个字段有一定的关系

6．下述关于数据库系统的叙述中正确的是（　　）。

A．数据库系统减少了数据冗余

B．数据库系统避免了一切冗余

C．数据库系统中数据的一致性是指数据类型一致

D．数据库系统比文件系统能管理更多的数据

7．数据库 DB、数据库系统 DBS、数据库管理系统 DBMS 之间的关系是（　　）。

A．DB 包含 DBS 和 DBMS　　B．DBMS 包含 DB 和 DBS

C．DBS 包含 DB 和 DBMS　　D．没有任何关系

8．在数据管理技术的发展过程中，可实现数据共享的是（　　）。

A．人工管理阶段　　B．文件系统阶段

C．数据库系统阶段　　D．系统管理阶段

9．1970 年，美国 IBM 公司研究员 E.F.Codd 提出了数据库的（　　）。