高等学校计算机应用规划教材

数据库程序设计

严　南　宋　容　主　编
袁连海　姚　捃　副主编

清华大学出版社

北　京

内 容 简 介

本书以培养复合型应用人才为目标，贴近全国计算机等级考试大纲，详细介绍 Access 2016 的主要功能和使用方法，其中在第 9 章加入了等级考试大纲要求的公共基础知识。全书共 9 章，主要内容包括数据库系统基础知识、数据库和表的基本操作、查询及其应用、窗体的创建与使用、报表的操作、宏的创建与使用、模块与 VBA 程序设计基础、VBA 数据库编程和公共基础知识。

本书结构严谨，知识点全面，通俗易懂，注重实用性和可操作性。全书理论和实践联系紧密，实例丰富，读者可以边学习、边实践，从最基本的建立数据库和表开始逐步建立数据库中的各种对象，掌握模块和程序设计基础知识，并在学习过程中形成计算机逻辑思维能力。

本书可作为高等院校非计算机专业数据库程序设计及相关课程的教材，也可作为全国计算机等级考试二级 Access 数据库程序设计的自学参考用书。

本书配套的电子课件、实例源文件和习题答案可以到 http://www.tupwk.com.cn/downpage 网站下载，也可以扫描前言中的二维码下载。

图书在版编目(CIP)数据

数据库程序设计 / 严南， 宋容主编.—北京：清华大学出版社，2021.2
(高等学校计算机应用规划教材)
ISBN 978-7-302-57560-3

Ⅰ. ①数… Ⅱ. ①严… ②宋… Ⅲ. ①关系数据库系统—高等学校—教材 Ⅳ. ①TP311.138

中国版本图书馆 CIP 数据核字(2021)第 027902 号

责任编辑：胡辰浩
装帧设计：孔祥峰
责任校对：成凤进
责任印制：

出版发行：清华大学出版社
网　　址：http://www.tup.com.cn，http://www.wqbook.com
地　　址：北京清华大学学研大厦 A 座　　邮　　编：100084
社 总 机：010-62770175　　邮　　购：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质 量 反 馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
印 装 者：
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张：**14.75　　**字　　数：**377 千字
版　　次：2021 年 3 月第 1 版　　**印　　次：**2021 年 3 月第 1 次印刷
印　　数：1～2000
定　　价：69.00 元

产品编号：090348-01

前　言

本书以 Microsoft Access 2016 为操作平台。Access 2016 是微软公司推出的一款易学、功能完备的数据库管理系统软件，其主要功能是管理和应用数据库。与 Access 的以前版本相比，Access 2016 除了继承和发扬了以前版本的功能强大、界面友好、易学易用的优点之外，在界面的易用性和支持网络数据库方面进行了很大改进。

本书介绍关系数据库管理系统的基本知识和 Access 数据库系统的主要功能。本书理论论述通俗易懂、重点突出、循序渐进，案例操作步骤清晰、简明扼要、图文并茂。本书强调理论知识与实际应用的有机结合，正文讲解与课后练习呼应补充，每章的课后习题大部分是全国计算机二级考试理论部分真题。除第 1 章外，每章都有实训项目，实训项目内容基本是全国计算机二级考试操作部分真题。

本书共分为 9 章，由浅入深地对 Access 2016 进行了详细的讲解并以示例为引导介绍 Access 的各项功能尤其是它的新功能，同时每个示例都配有图片。本书注重实践，读者按照示例一步一步去做即可掌握 Access 的基本内容和常用功能，也可以完成一个基本的数据库应用开发。

本书提供了丰富的实训操作和大量习题，各章后均有“小结”，以总结教学重点和教学要点。为了方便教学，本书为教师提供了电子课件、习题答案以及操作实训所用到的全部素材。

本书的第 1 章由袁连海编写，第 2、3、5 章由宋容编写，第 4、7、8 章由严南编写，第 6、9 章由姚捃编写。全书由严南统稿和审定。在本书的编写和出版过程中，得到了成都理工大学工程技术学院和清华大学出版社的大力支持，在此表示衷心感谢。

由于编者水平有限，书中不妥之处在所难免，敬请读者批评指正。我们的电话是 010-62796045，邮箱是 huchenhao@263.net。

本书配套的电子课件、实例源文件和习题答案可以到 http://www.tupwk.com.cn/downpage 网站下载，也可以扫描下方的二维码下载。

作　者

2020 年 11 月

目　录

第 1 章

数据库概述

当今社会是信息爆发式增长的社会。信息社会离不开数据。人们日常生活中需要处理许许多多的数据，这些数据的收集、处理、存储、发布以及对数据的挖掘利用，都会使用数据库。数据库技术产生于 20 世纪 60 年代末，是数据管理的最新技术，是计算机科学的重要研究分支。在数据库领域，出现了很多杰出的计算机科学家。信息社会信息化程度的高低依赖于数据库技术的发展水平。

本章首先介绍数据库的基本概念，对数据库、数据库系统和数据库管理系统等进行介绍；接着介绍数据管理技术发展的三个阶段的特点，介绍数据模型和关系运算；最后对 E-R 模型以及规范化理论进行介绍。

1.1 数据库的基本概念

数据库技术是信息系统的核心和基础，它的出现极大地促进了计算机应用向各行各业的渗透。人类对数据可以进行数据处理和数据管理。数据处理是对各种形式的数据进行收集、存储、加工和传输等活动的总称。数据的收集、分类、组织、编码、存储、检索、传输和维护等环节是数据处理的基本操作，称为数据管理。数据管理是数据处理的核心问题。一个国家数据库的建设规模、数据库信息量的大小和使用频度已成为衡量一个国家信息化程度的重要标志。

1.1.1 信息与数据库

什么是数据呢？数据(Data)是描述事物的符号记录，是数据库中存储的基本对象。数据分为数值数据和非数值数据，早期的计算机系统主要用于科学计算，处理的数据是整数、实数、浮点数等传统数学中的数据，现代计算机能存储和处理的对象十分广泛，表示这些对象的数据也越来越复杂。例如，王老师的年龄是 50 岁，某件商品的价格是 10.50 元，小王买的书的数量是 20 本，这些数据就是数值数据；而一个人的姓名是李平、性别是男、学号是 201920112020，这些数据是非数值型数据。数据包括文本(如表示姓名、性别、学号的数据)、数字(如年龄、价格、数量)、图形、图像、声音等。数据的解释是指对数据含义的说明，数据的含义称为数据的语义，数据与其语义是不可分的，例如，数据(李平，1972，1992)表示什么呢？如果语义是(学生姓名、出生年份、入学年份)，则从这个语义我们知道学生李平是 1972 年出生，入学年份是

1992 年；如果语义是(教师姓名、出生年份、参加工作年份)，则从这个语义我们知道教师李平是 1972 年出生，参加工作年份是 1992 年。再例如，200 这个数字可以表示一件物品的价格是 200 元，也可以表示一个专业的学生人数有 200 人，还可以表示一袋洗衣粉的重量是 200 克。

信息和数据有什么关系呢？为什么我们经常说信息社会而不说数据社会？信息是现实世界事物的存在方式或运动状态的反映。或认为，信息是一种已经被加工为特定形式的数据。信息的主要特征如下。

- 信息的传递需要物质载体，信息的获取和传递要消费能量。
- 信息可以感知；信息可以存储、压缩、加工、传递、共享、扩散、再生和增值。
- 数据是信息的载体和具体表现形式，信息不随数据形式的变化而变化。

数据有文字、数字、图形、声音等表现形式。例如，要将 2019 年 9 月 1 日下午 2 点在 1103 房间开会的信息通知某个同学，可以通过发送短消息(文本)或者打电话(语音)等数据形式发布。一般情况下将数据与信息作为一个概念而不加以区分。

数据库(Database，DB)是长期存储在计算机内的、有组织的、可共享的大量数据的集合。数据库具有较小的冗余度，较高的数据独立性和易扩展性，因为数据库中的数据是按某种数据模型进行组织的，数据存放在辅助存储器上，而且数据可被多个用户同时使用。所以，数据库中的数据按一定的数据模型组织、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，并可为各种用户共享。数据库技术所研究的问题是如何科学地进行数据管理，这就离不开一个重要的系统软件：数据库管理系统。

1.1.2 数据库管理系统

数据库管理系统(Database Management System，DBMS)是用来对数据库进行高效管理的系统软件。数据库管理系统是用户与操作系统之间的一层数据管理软件，目的是科学地组织和存储数据、高效地获取和维护数据。通过数据库管理系统，人们可以方便地对数据库中的数据进行收集、存储、操作和维护。数据库管理系统的主要功能包括以下几个方面：

- 数据定义功能
- 数据操纵功能
- 数据库的运行管理功能
- 数据库的建立和维护功能

数据库管理系统是维护和管理数据库的软件，是数据库与用户之间的接口。作为数据库的核心软件，提供建立、操作、维护数据库的命令和方法。DBMS 是一个大型的、复杂的软件系统，是计算机中的基础软件。目前，专门研究 DBMS 的厂商及研制的 DBMS 产品很多。比较流行的有美国 IBM 公司的 DB2 关系数据库管理系统和 IMS 层次数据库管理系统、美国 Oracle 公司的 Oracle 关系数据库管理系统、Sybase 公司的 Sybase 关系数据库管理系统、美国微软公司的 SQL Server 关系数据库管理系统以及目前十分流行的开源关系数据库管理系统 MySQL 等。

数据库管理系统是操纵和管理数据库的一组软件，是数据库系统(DBS)的重要组成部分。不同的数据库系统都配有各自的 DBMS，而不同的 DBMS 各支持一种数据库模型，虽然它们的功能强弱不同，但大多数 DBMS 的构成相同，功能相似。一般来说，DBMS 具有定义、建立、

维护和使用数据库的功能，通常由三部分构成：数据描述语言及其翻译程序、数据操纵语言及其处理程序和数据库管理的例行程序。

Access 2016 是一个关系数据库管理系统，它是微软开发的 Office 2016 套件中的组件之一，比较适合中小型企业。特点是用户界面友好，操作简单，面向对象，事件驱动，支持对多媒体数据的管理，内置大量的函数和宏，支持 VBA 编程。

1.1.3　数据库系统

数据库系统(Data Base System，DBS)是指在计算机系统中引入数据库后的系统，一般由数据库、数据库管理系统(及其开发工具)、应用系统、数据库管理员构成。数据库系统和数据库是两个不同的概念，数据库系统是一个人机系统，例如航空售票系统、公交信息查询系统和旅游信息系统等，数据库是数据库系统的一个组成部分。但在日常工作中，人们常常把数据库系统简称为数据库。读者需要能够从人们讲话或文章的上下文中区分“数据库系统”和“数据库”，不要将二者混淆。

使用数据库系统的好处是由数据库管理系统的特点或优点决定的。使用数据库系统的好处很多，例如，可以大大提高应用开发的效率，方便用户的使用，减轻数据库系统管理人员维护系统的负担等。使用数据库系统可以大大提高应用开发的效率，因为在数据库系统中应用程序不必考虑数据的定义、存储和存取路径，这些工作都由 DBMS 来完成。用一个通俗的比喻，使用了 DBMS 就如有了一个好参谋、好助手，许多具体的技术工作都由这个助手来完成。开发人员就可以专注于应用逻辑的设计，而不必为数据管理的许多复杂细节操心。还有，当应用逻辑改变，数据的逻辑结构也需要改变时，由于数据库系统提供了数据与程序之间的独立性，数据逻辑结构的改变是数据库管理员(DBA)的责任，开发人员不必修改应用程序，或者只需要修改很少的应用程序，从而既简化了应用程序的编制，又大大减少了应用程序的维护和修改。

使用数据库系统还可减轻数据库系统管理人员维护系统的负担。因为 DBMS 在数据库建立、运用和维护时对数据库进行统一的管理和控制，包括数据的完整性、安全性、多用户并发控制、故障恢复等，都由 DBMS 执行和控制。总之，使用数据库系统的优点是很多的，既便于数据的集中管理，控制数据冗余，提高数据的利用率和一致性，又有利于应用程序的开发和维护。数据库(DB)、数据库系统(DBS)和数据库管理系统(DBMS)三者之间的关系是：DBS 包括 DB 和 DBMS。

图 1-1 所示为数据库、数据库管理系统和数据库系统这三者的关系。

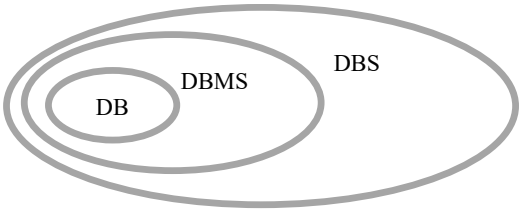

图 1-1　DBS、DBMS 和 DB 的关系

1.2 数据管理技术的发展阶段

数据管理是对数据进行分类、组织、编码、存储、检索和维护的过程，是数据处理的核心问题。

推动人类数据管理技术发展的动力包括应用需求的推动以及计算机硬件的发展和计算机软件的发展。人类数据管理技术的发展经历了以下三个阶段：

- 人工管理阶段(20 世纪 40 年代中期到 20 世纪 50 年代中期)
- 文件系统阶段(20 世纪 50 年代末到 20 世纪 60 年代中期)
- 数据库系统阶段(20 世纪 60 年代末到现在)

结合当时所处的历史阶段，以及软硬件发展水平，同学们不难理解数据管理所具有的特点以及制约条件。例如，在人工管理阶段，由于没有操作系统和存储设备，人类对数据的管理只能通过人工管理。随着操作系统的出现和随机存储设备的发展，可以通过文件来管理数据。

1.2.1 人工管理阶段

在计算机出现的初期，由于当时的软件和硬件的限制，人们对数据管理采用人工管理方式，从 20 世纪 40 年代中期到 50 年代中期将近十年的时间内，都是以人工管理方式对数据进行管理的。

当时的计算机主要用来进行科学计算，计算机没有操作系统，也没有直接存储设备。这些条件限制了人们只能采用人工方式管理数据。

这个阶段的数据管理者是用户(也就是程序员)，由于没有磁盘和磁带，数据不能保存在存储设备里面。数据面向的对象是某个特定的应用程序(简称应用)。例如，统计某个地区的人口信息，处理的数据只是针对统计程序。数据的共享程度低，几乎没有共享性，数据的冗余度很大；数据没有独立性，完全依赖于某个应用程序。当然，这个阶段的数据是没有结构的，程序员编写的应用程序自行控制数据。

1.2.2 文件系统阶段

随着计算机软硬件技术的发展，在 20 世纪 60 年代初期到 60 年代中后期，出现了随机存储磁盘，软件方面出现了操作系统，操作系统里面有文件系统，计算机应用不仅用于科学计算，还用来进行数据管理，操作系统除了有批处理系统，也有联机处理系统。这些技术允许人们使用文件系统来管理数据。

文件系统阶段的数据管理具有以下特点：通过文件系统来管理数据，数据可长期保存在设备上；数据依然是面向某一特定的应用程序；数据的共享性比较差，数据冗余度大；整体上看，数据没有结构，但记录内有结构；数据的独立性仍然较差，数据的逻辑结构改变必须修改应用程序，应用程序自己控制数据；文件中记录内是有结构的，数据的结构是由程序定义和解释的，数据只能是定长的，可以间接实现数据变长要求，但访问相应数据的应用程序复杂了；文件间是独立的，因此数据整体无结构；可以间接实现数据整体的有结构，但必须在应用程序中描述数据间的联系；数据的最小存取单位是记录。

采用文件系统管理数据相对于人工管理具有很大进步和优点，但这种管理方式依然有以下几个缺陷：

- 数据冗余大；
- 数据不一致性；
- 数据独立性较差。

例如，学校教务处、财务处、学生处几个部门分别开发的应用程序都在文件里面定义了学生信息，如姓名、联系电话、家庭住址等，这就是数据冗余。如果某个学生的家庭住址改变了，就要去修改这三个部门的文件中的学生的家庭住址信息，否则就会引起同一学生的数据不同部门中不一致，产生上述问题的原因是这三个部门的应用程序的文件中关于学生的数据没有联系，是相互独立的。

有些应用适用于文件系统而不是数据库系统，例如对于数据的备份、应用程序使用过程中产生的临时数据，一般使用文件系统比较合适。对于早期功能比较简单、比较固定的应用系统，一般适合采用文件系统。而目前几乎所有企业或部门的信息系统都是以数据库系统为基础的，数据的存储都使用数据库。例如，一个工厂的管理信息系统(其中包括许多子系统，如设备管理系统、物资采购系统、库存管理系统、作业调度系统、人事管理系统等)，学校的学籍管理系统，人事管理系统，图书馆的图书管理系统等，都比较适合采用数据库系统。

1.2.3　数据库系统阶段

数据库系统阶段比文件系统阶段更为高级，它可以解决多用户、多应用共享数据的需求，使得数据尽可能面向更多的应用。数据库系统阶段不再使用人工和文件来管理数据，而使用专门的数据管理软件——数据库管理系统来管理数据。数据库系统阶段与文件系统阶段最大的差别在于数据的结构化。

与文件系统阶段相比，数据库系统阶段主要有以下三个优点。

第一个优点是数据结构化。数据结构化是数据库的主要特征之一，数据库系统实现整体数据的结构化是数据库系统与文件系统的本质区别。在数据库系统阶段，数据不再针对某一特定应用设计，而是面向全组织(单位)，数据库系统阶段的数据具有整体的结构化特点。在文件系统中，数据的存取单位只有一个——记录，如一个学生的完整记录(学过 C 语言的同学可以思考如何用 C 语言编写一个电话簿来管理朋友的信息)。不仅数据是结构化的，而且数据的存取数量(即一次可以存取数据的大小)也很灵活，可以小到某一个数据项(如一个学生的学号)，大到一组记录(成千上万个学生记录)。

第二个优点是数据的共享性程度高，冗余度小，容易扩充等。由于数据面向整个系统，是有结构的数据，不仅可以被多个应用共享使用，而且容易增加新的应用，数据库阶段的数据不再面向某个应用程序，而面向整个系统，因此可以被多个用户、多个应用以多种不同的语言共享使用。这就使得数据库系统弹性大，易于扩充。数据共享可以大大减少数据冗余，节约存储空间，还能避免数据之间的不相容性与不一致性。在文件系统阶段，数据是面向某个应用程序而设计的，也就是数据结构是针对某个具体应用设计的，数据只被这个应用程序或应用系统使用，可以说数据是某个应用的私有资源，数据库阶段的系统容易扩充也容易收缩，即应用增加或减少的时候不需要修改整个数据库的结构，只需要做很少的改动。可以取整体数据的各种子

集用于不同的应用系统，应用需求发生改变时，只需要重新选取不同的子集或者添加一部分数据，即可满足新的需求。

第三个优点是数据独立性高。数据独立性是数据库系统的最重要特点之一，它使数据能独立于应用程序。数据独立性包括数据的物理独立性和数据的逻辑独立性。数据库管理系统的三级模式结构(外模式、模式和内模式)和二级映射功能保证了数据库中的数据具有很高的物理独立性和逻辑独立性。物理独立性是指用户的应用程序与存储在磁盘上的数据库中的数据是相互独立的，即数据在磁盘上怎样存储由 DBMS 管理，用户程序不需要了解，应用程序要处理的只是数据的逻辑结构，这样当数据的物理存储改变了，应用程序不用改变。逻辑独立性是指用户的应用程序与数据库的逻辑结构是相互独立的，即当数据的逻辑结构改变时，用户程序也可以不变。数据与程序的独立，把数据的定义从程序中分离出去，加上数据的存取又由 DBMS 负责，从而简化了应用程序的编制，大大减少了应用程序的维护和修改。可以说数据处理的发展史就是数据独立性不断进化的历史。在手工管理阶段，数据和程序完全交织在一起，没有独立性可言，数据结构做任何改动，应用程序也需要做相应的修改。文件系统出现后，虽然将两者分离，但实际上应用程序中依然要反映文件在存储设备上的组织方法、存取方法等物理细节，因而只要数据做了任何修改，程序仍然需要做改动。而数据库系统的一个重要目标就是使程序和数据真正分离，使它们能独立发展。

在数据库系统阶段，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了统一的数据定义、数据控制、安全机制以及一系列备份和恢复机制。另外，数据库管理系统提供数据库的共享机制，允许多个用户同时存取数据库中的数据甚至可以同时存取数据库中同一个数据。为此，DBMS 必须提供统一的数据控制功能，包括并发控制、数据的完整性检查、数据的安全性保护和数据库恢复。并发控制，对多用户的并发操作加以控制和协调，保证并发操作的正确性；数据的完整性检查，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；数据的安全性保护，保护数据以防止不合法的使用造成的数据的泄密和破坏；数据库恢复，当计算机系统发生硬件故障、软件故障，或者由于操作员的失误以及故意的破坏影响数据库中数据的正确性，甚至造成数据库部分或全部数据的丢失时，能将数据库从错误状态恢复到某一已知的正确状态(亦称为完整状态或一致状态)。

1.3 数据模型

数据模型是数据库中用来对现实世界进行抽象的工具，是数据库中用于提供信息表示和操作手段的形式构架。将现实世界转换成机器世界涉及几个概念。例如，要采用计算机来实现教务管理，需要经过几次建模，将现实世界转换成信息世界使用的模型称为概念模型，概念模型是数据库设计者交流的工具。概念模型实际上是现实世界到机器世界的一个中间层次。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，是数据库设计人员进行数据库设计的有力工具，也是数据库设计人员和用户之间进行交流的语言。

建立概念模型后，需要将概念模型转换成某种具体数据库系统支持的模型，在机器世界使用的模型称为数据模型。概念模型中最常用的是实体联系模型(E-R 模型)，概念模型的目的是

根据需求分析得到概念模型(即 E-R 图)。E-R 图是数据库设计人员之间交流的工具，与具体的 DBMS 无关。接下来是将 E-R 图转换为某一种数据模型，数据模型也与 DBMS 相关。

图 1-2 表示将现实世界抽象成机器世界需要进行的两次抽象。第一次抽象是从现实世界到信息世界的抽象，得到概念模型，其中最常见也最容易理解的是 E-R 图；第二次抽象是将概念模型(如 E-R 图)转换成机器世界的数据模型(如最常用的关系模型)。

图 1-2　数据抽象

1.3.1　数据模型的分类

数据模型是数据库系统的基础，任何数据库管理系统都要按照一定的方式组织数据，数据模型是数据库管理系统用来对现实世界进行抽象的工具，是数据库中用于提供信息表示和操作手段的形式构架。一般来说，数据模型是严格定义的概念的集合。这些概念精确描述了系统的静态特性、动态特性和完整性约束条件。数据模型包括数据结构、数据操作和完整性约束三个要素。

- 数据结构：从静态特性描述数据，是研究对象类型的集合。
- 数据操作：描述可以对数据库中各种对象进行的什么操作，是操作的集合，包括操作及有关的操作规则，是对系统动态特性的描述。
- 完整性约束条件：约束条件是一组完整性规则的集合。完整性规则是给定的数据模型中数据及其联系所具有的制约和依存规则，用以限定符合数据模型的数据库状态以及状态的变化，以保证数据的正确、有效、相容。

任何一个数据库管理系统都以某种数据模型为基础，或者说支持某一个数据模型。数据库系统中，模型有不同的层次。根据模型应用的目的不同，可以将模型分成两类或者两个层次：一类是概念模型，按用户的观点来对数据和信息建模，用于信息世界的建模，强调语义表达能力，概念简单清晰；另一类是数据模型，按计算机系统的观点对数据建模，用于机器世界，人们可以用它定义、操纵数据库中的数据，一般需要有严格的形式化定义和一组严格定义了语法和语义的语言，并有一些规定和限制，便于在机器上实现。数据库管理系统常用的数据模型包括层次数据模型、网状数据模型和关系数据模型。

1. 层次数据模型

层次数据模型采用树<层次>结构来组织数据，层次数据模型的图形表示是一棵倒立生长的树，由数据结构中树(或者二叉树)的定义可知，每棵树都有且仅有一个根节点，其余的节点都是非根节点。每个节点表示一个记录类型对应于实体的概念，记录类型的各个字段对应实体的各个属性。各个记录类型及其字段都必须记录。

层次数据模型具有以下特点。

(1) 整个模型中有且仅有一个节点没有父节点，其余的节点必须有且仅有一个父节点，但

是所有的节点都可以不存在子节点。

(2) 所有的子节点不能脱离父节点而单独存在。也就是说，如果要删除父节点，那么父节点下面的所有子节点都要同时删除，但是可以单独删除一些叶子节点。

(3) 每个记录类型有且仅有一条从父节点通向自身的路径。

图 1-3 以某所大学某个系的组织结构为例，说明层次数据模型的结构。

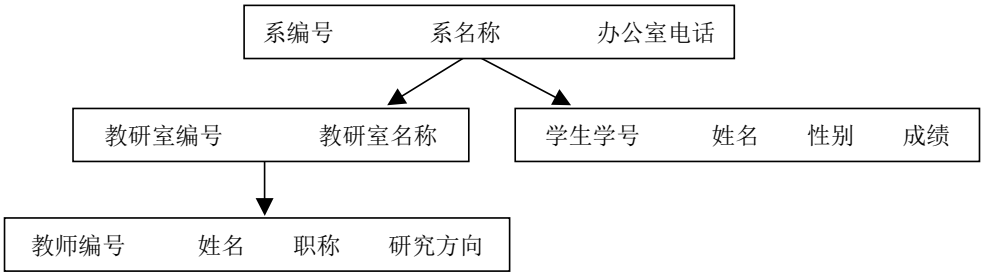

图 1-3 学校层次数据模型

(1) 系是根节点，其属性包括系编号、系名称和办公室电话。

(2) 教研室和学生分别构成了记录类型系的子节点，教研室的属性有教研室编号和教研室名称，学生的属性包含学生学号、姓名、性别和成绩。

(3) 教师是教研室这一实体的子节点，其属性有教师编号、姓名、职称和研究方向。

层次数据模型采用树结构来组织数据，因此层次数据模型具有以下优点。

(1) 模型简单，层次数据模型的结构简单、清晰、明朗，可以很容易看到各个实体之间的联系，对具有一对多层次关系的部门描述非常自然、直观，容易理解，这是层次数据库的突出优点。

(2) 用层次数据模型的应用系统性能好，查询效率较高，在层次数据模型中，节点的有向边表示了节点之间的联系，在 DBMS 中如果有向边借助指针实现，那么依据路径很容易找到待查的记录；操作层次数据类型的数据库语句比较简单，只需要几条语句就可以完成数据库的操作，特别是对于那些实体间联系是固定的且预先定义好的应用，采用层次数据模型来实现，其性能优于关系数据模型。

(3) 层次数据模型提供了良好的数据完整性支持，正如上面所说，如果要删除父节点，那么其下的所有子节点都要同时删除，如图 1.3 中，如果想要删除教研室，则其下的所有教师都要删除。

层次数据模型具有以下缺点。

(1) 结构缺乏灵活性，层次数据模型只能表示实体之间的 1∶n 的关系，不能表示 m∶n 的复杂关系，因此现实世界中的很多模型不能通过该模型方便地表示。现实世界中很多联系是非层次性的，如多对多联系、一个节点具有多个双亲等，层次数据模型不能自然地表示这类联系，只能通过引入冗余数据或引入虚拟节点来解决。

(2) 对插入和删除操作的限制比较多。

(3) 查询子女节点必须通过双亲节点。由于查询节点的时候必须知道其双亲节点，因此限制了对数据库存取路径的控制。

2. 网状数据模型

网状数据模型采用有向图表示实体和实体之间的联系。网状数据模型可以被看成放松层次数据模型的约束性的一种扩展。网状数据模型中所有的节点允许脱离父节点而存在。也就是说，在整个模型中允许存在两个或多个没有根节点的节点，同时也允许一个节点存在一个或者多个父节点，成为一种网状的有向图。因此，节点之间的对应关系不再是 1∶n，而是一种 m∶n 的关系，从而克服了层次数据模型的缺点。

图 1-4 以教务管理系统为例，说明了院系的组成中，教师、学生、课程之间的关系。可以从图中看出课程(实体)的父节点为专业、教研室、学生。以课程和学生之间的关系来说，是一种 m∶n 的关系。也就是说，一个学生能够选修多门课程，一门课程也可以被多个学生同时选修。网状数据模型中的每个节点表示一个实体，节点之间的有向线段表示实体之间的联系。

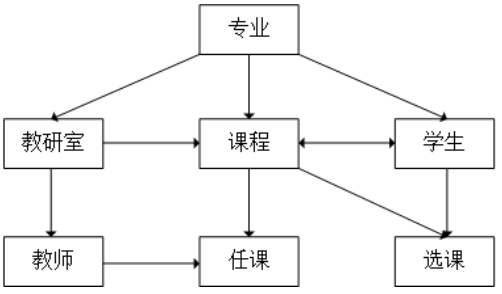

图 1-4　网状数据模型

网状数据模型具有以下优点。

(1) 网状数据模型可以很方便地表示现实世界中很多复杂的关系，能够更直接地描述现实世界，如一个节点可以有多个双亲。

(2) 修改网状数据模型时，没有层次数据模型那么多严格限制，可以删除一个节点的父节点而依旧保留该节点；也允许插入一个没有任何父节点的节点，这样的插入在层次数据模型中是不被允许的，除非首先插入的是根节点。

(3) 实体之间的关系在底层中可以通过指针实现，因此在这种数据库中执行操作的效率较高。

当然，网状数据模型也有很多缺点。网状数据模型结构复杂，使用不容易，随着应用环境的扩大，数据结构变得越来越复杂，不利于最终用户掌握；数据的插入、删除牵动的相关数据太多，不利于数据库的维护和重建。由于记录之间的联系是通过存取路径实现的，应用程序在访问数据时必须选择适当的存取路径。因此，用户必须了解系统结构的细节，加重了编写应用程序的负担。网状数据模型数据彼此关联较大，该模型其实是一种导航式的数据模型结构，不仅要说明要对数据做些什么，还要说明操作的路径。

3. 关系数据模型

关系数据模型使用关系(二维表)来表示实体和实体之间的联系。关系数据模型对应的数据库自然就是关系数据库，支持关系数据模型的数据库管理系统称为关系数据库管理系统。这是目前应用最多的数据库。同理，使用层次数据模型的数据库称为层次数据库，而使用网状数据模型的数据库称为网状数据库。

关系数据库是目前最流行的数据库，同时也是被普遍使用的数据库，如 MySQL、SQL Server、Oracle 等都是流行的关系数据库。

在关系数据模型中，无论是实体还是实体之间的联系都被映射成统一的关系：一张二维表，在关系数据模型中，操作的对象和结果都是一张二维表。关系数据库可用于表示实体之间多对多的关系，只是此时需要借助第三张表来实现多对多的关系，例如，学生选课系统中学生和课程之间的联系是一种多对多的关系，这种多对多的联系也是转换成二维表(关系)。例如，选课系统涉及三张表，分别是学生表、课程表和选课表，而选课表将学生和课程联系起来。关系数据模型的关系必须是规范化的关系，即每个属性是不可分割的实体，不允许表中嵌套另一张表。

1.3.2 关系数据模型

关系数据模型由关系数据结构、关系操作的集合和关系完整性约束三部分组成。从用户观点来看，关系数据模型中，逻辑数据结构是一张简单的二维表，它由行和列组成。例如，图 1-5 所示的日常生活中常见的二维表就是关系。

学号	姓名	性别	年龄
2018202011	李平	男	19
2018202012	王梅	女	20
2018202013	董东	男	18
2018202014	王芳	女	19

图 1-5　学生关系

在关系数据模型中，有一些概念需要理解并掌握。

1. 关系

一个关系就是一张二维表。通常将一个没有重复行、重复列，并且每个行列的交叉点只有一个基本数据的二维表格看成一个关系，每个关系都有一个关系名。

例如，图 1-5 这张二维表是一个关系，关系名叫学生关系。

2. 元组

二维表除了第一行之外的每一行在关系中称为一个元组。在 Access 中，一个元组对应表中的一条记录。

例如，图 1-5 中第二行在关系中成为元组，如果在二维表中称为记录。

3. 属性

二维表的每一列在关系中称为属性。每个属性都有一个属性名，一个属性在其每个元组上的值称为属性值。一个属性包括多个属性值。在 Access 中，一个属性对应二维表中的一个字段(Field)，属性名对应字段名，属性值对应各个记录的字段值。

例如，图 1-5 中学号这列称为属性或者字段，学号称为属性名或者字段名，“2018202011”“2018202012”“2018202013”“2018202014”称为属性值或者字段值。

4. 域

属性或者字段的取值范围称为域。域作为属性值的集合，其类型与范围由属性的性质及其所表示的意义具体确定。同一属性只能在相同域中取值。例如，图 1-5 中的“性别”属性的域是“男”或“女”。

5. 关键字

在关系数据模型中，能唯一标识关系中不同元组的属性或属性组合，称为该关系的一个关键字。单个属性组成的关键字称为单关键字，多个属性组成的关键字称为组合关键字。

关系中能够成为关键字的属性或属性组合可以有多个。凡是在关系中能够唯一区分、确定不同元组的属性或属性组合，均称为候选键或候选关键字。在候选关键字中选定一个并且只能一个作为该关系的主关键字，简称主键或主码(Primary Key，PK)。关系中的主关键字是唯一的。例如，图 1-5 中，假设学号、姓名这两个属性没有重复值，就可以把学号和姓名当成候选关键字，学号当成主关键字。一个表中主关键字只能有一个。

还有一种关键字称为外部关键字。一个关系中某个属性或属性组合并非关键字，但却是另一个关系的主关键字，称此属性或属性组合为本关系的外部关键字。它是关系之间联系的纽带，关系之间的联系是通过外部关键字实现的，本书 2.5 节将有详细的阐述。

6. 关系模式

对关系的描述称为关系模式，其表示格式如下：

```
关系名(属性名 1，属性名 2，…，属性名 n)
```

关系既可以用二维表格来描述，也可以用数学形式的关系模式来描述。一个关系模式对应一个关系的结构。在 Access 中，这就是表的结构，其表示如下：

```
表名(字段名 1，字段名 2，…，字段名 n)
```

例如，图 1-5“学生关系”表对应的关系模式可表示为：学生关系(学号，姓名，性别，年龄)。

在关系数据库中，关系具有以下性质：

- 所有的属性都是原子属性。
- 元组的顺序无关紧要，即元组的次序可以任意交换。
- 属性的顺序是非排序的，即它的次序可以任意交换。
- 同一属性名下的各个属性值(同列)是同类型数据，且来自同一个域。
- 关系中没有重复元组，任意元组在关系中都是唯一的。
- 不同属性必须具有不同的属性名，不同属性可来自同一个域。

绝大多数数据库系统在总的体系结构上都具有三级模式的特征。三级模式是对数据的三个抽象级别。图 1-6 所示是数据库的三级模式。

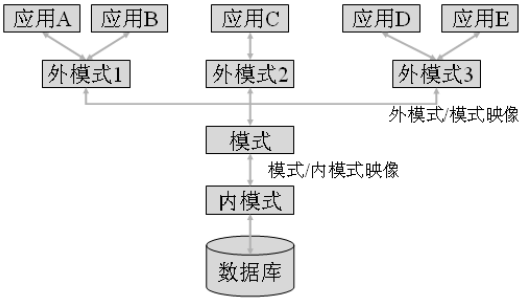

图 1-6　数据库的三级模式

根据不同角度理解数据库的数据，体系结构分为三级模式：外模式、模式和内模式。数据库管理系统在三级模式之间提供的两层映像(外模式/模式映像、模式/内模式映像)保证了数据库系统中的数据能够具有较高的逻辑独立性和物理独立性。

外模式又称为用户模式或者子模式，它是数据库用户(包括应用程序员和最终用户)能够看见和使用的局部数据的逻辑结构和特征的描述，是数据库用户的数据视图，是与某一应用有关的数据的逻辑表示。例如，关系数据库的视图就是外模式。外模式保证了数据库的安全性。每个用户只能看见和访问所对应的外模式中的数据，数据库中的其余数据是不可见的。

模式又称为逻辑模式，是数据库中全体数据的逻辑结构和特征的描述，是所有用户的公共数据视图。一个数据库只有一个模式。在定义模式时不仅要定义数据的逻辑结构，例如，数据记录由哪些数据项构成，数据项的名称、类型、取值范围等，而且要定义数据之间的联系，定义与数据有关的安全性、完整性要求。

内模式又称为存储模式或者物理模式。一个数据库只有一个内模式。它是数据物理结构和存储方式的描述，是数据在数据库内部的表示方式。例如，数据是否加密，是否压缩存储，数据的存储记录结构有何规定等。

正是因为数据库具有三级模式和两级映射，从而保证了数据的逻辑独立性和物理独立性。数据的逻辑独立性是指当数据库的模式发生改变时，应用程序不需要改变。例如，在数据库中增加新的表、新的字段，改变某个表中属性的数据类型或者长度等，改变了模式，则数据库管理员对各个外模式/模式的映像做相应改变，可以使外模式保持不变，也就是应用程序不需要改变。这是因为应用程序是依据数据的外模式编写的，从而应用程序不必修改，保证了数据与程序的逻辑独立性，简称数据的逻辑独立性。数据与程序的物理独立性是指当数据库的存储结构发生改变，数据库管理员只需要对模式/内模式之间的映像做相应改变，从而可以使模式不必改变，因此应用程序也不用改变，这样保证了数据与程序的物理独立性，简称数据的物理独立性。

关系的完整性规则包括实体完整性、参照完整性和用户自定义完整性三种。

实体完整性规则的意思是关系中元组在主码上不能相同或者不能为空值(NULL，不确定的意思)。如果出现空值，那么关键值就起不了唯一标识元组的作用。例如，在输入学生表的数据时，学号为主码，因此学号字段不能不输入(为空)，也不能相同。

参照完整性规则指外码的取值要么为空，要么取主码表中的值，而不能取其他的值。例如，学生表和班级表可以用下面的关系模式表示，其中主码用下画线标识：

学生(学号，姓名，性别，班级号，年龄，籍贯)

班级(班级号，班级名，班主任姓名)

这两个关系之间存在着属性的引用，即学生关系的班级号引用了班级关系的班级号。显然，学生关系中的班级号的取值要么为空值(也就是这个学生的班级不确定)，要么必须是确实存在的班级表的班级号，而不能是其他的值，即班级关系中有该班级的记录。也就是说，学生关系中的某个属性的取值需要参照班级关系的属性取值。

用户自定义完整性规则是针对某一具体数据的约束条件，由具体应用环境决定。它反映某一具体应用所涉及的数据必须满足的语义要求。例如，年龄的取值只能是大于 0 的正整数，而不能是负数。

1.4 关系运算

关系数据库建立在关系代数理论的基础之上。有很多数据理论可以表示关系模型的数据操作，其中最著名的是关系代数与关系运算。关系运算的运算对象是关系，运算结果也是关系，在离散数学中，二元关系也属于特殊的集合，因此关系运算包括传统的集合运算和专门的关系运算两类。传统的集合运算是从关系的水平方向，即行的角度来进行的，主要是集合与集合之间的运算，包括并、交、差 、笛卡儿积；而专门的关系运算不仅涉及行，还涉及列，包括选择、投影、连接、除。

1.4.1 传统的集合运算

传统的集合运算是二目运算(又称二元运算)。以下运算用到的两个关系 R 和 S 均为 n 元关系，且相应的属性取自同一个域，如图 1-7 所示。

关系 R

姓名	年龄	性别
李平	20	男
王梅	21	女
袁弘	20	男

关系 S

姓名	年龄	性别
李平	20	男
柳军	22	男
张彤	20	女

图 1-7　关系 R 和 S

基本运算如下：

1. 并(Union)

关系 R 和 S 的并为：

$$R \cup S=\{t|t \in R \vee t \in S\}$$

其结果仍为 n 目关系。任取元组 t ，当且仅当 t 属于 R 或 t 属于 S 时，t 属于 R∨S。例如，上述集合 R 和 S 的并集结果如图 1-8 所示。

姓名	年龄	性别
李平	20	男
王梅	21	女
袁弘	20	男
柳军	22	男
张彤	20	女

图 1-8　R∪S

2. 交(Intersection)

关系 R 和 S 的交为：

$$R \cap S=\{t|t \in R \wedge t \in S\}$$

其结果仍为 n 目关系。任取元组 t，当且仅当 t 既属于 R 又属于 S 时，t 属于 R∩S。从集合论的观点分析，关系的交运算可表示为差运算：R∩S=R－(R－S)。例如，上述集合 R 和 S 的交集结果如图 1-9 所示。

姓名	年龄	性别
李平	20	男

图 1-9　R∩S

3. 差(Difference)

关系 R 和 S 的差为：

$$R-S=\{t|t \in R \neg\ t \in S\}$$

其结果仍为 n 目关系。任取元组 t，当且仅当 t 属于 R 且 t 不属于 S 时，t 属于 R-S。R−S 的元素是属于 R 不属于 S 的元组。例如，上述集合 R 和 S 的差集结果如图 1-10 所示。

姓名	年龄	性别
王梅	21	女
袁弘	20	男

图 1-10　R−S

4. 笛卡儿乘积(Cartesian Product)

设 R 为 m 目关系，S 为 n 目关系，则 R 和 S 的广义笛卡儿乘积为：

$$R \times S=\{t|t=<tr, ts> \wedge tr \in R \wedge ts \in S\}$$

其结果为 m+n 目关系。元组的前 m 列是关系 R 的一个元组，元组的后 n 列是关系 S 的一个元组。若 R 有 k_1 个元组，S 有 k_2 个元组，则 R×S 有 $k_1 \times k_2$ 个元组。

例如，R 和 S 关系的笛卡儿乘积如图 1-11 所示。

R.姓名	R.年龄	R.性别	S.姓名	S.年龄	S.性别
李平	20	男	李	20	男
李平	20	男	柳	22	男
李平	20	男	张	20	女
王梅	21	女	李	20	男
王梅	21	女	柳	22	男
王梅	21	女	张	20	女
袁弘	20	男	李	20	男
袁弘	20	男	柳	22	男
袁弘	20	男	张	20	女

图 1-11　关系的笛卡儿乘积 R×S

对关系 R 和 S 的并、交、差和笛卡儿乘积进行描述见图 1-12。

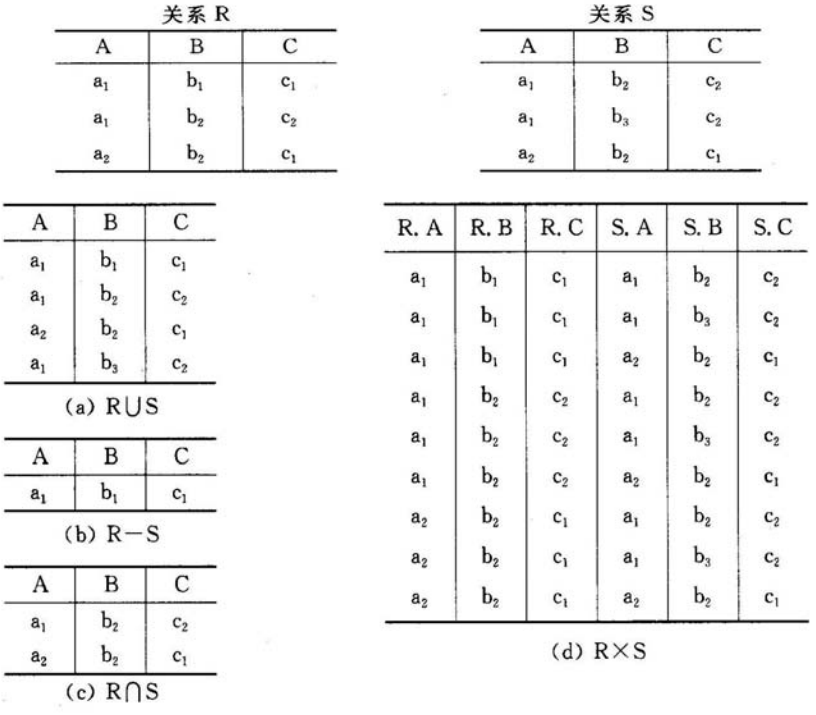

图 1-12　关系 R 和 S 的并、交、差和笛卡儿乘积

实际进行笛卡儿乘积运算时，可从 R 的第一个元组开始，依次与 S 的每一个元组组合，然后对 R 的下一个元组进行同样的操作，直至 R 的最后一个元组也进行完相同操作为止，即可得到 R×S 的全部元组。

1.4.2　专门的关系运算

专门的关系运算包括选择、投影、连接和除。前两个是一元操作，后两个为二元操作，我们重点对选择、投影和连接进行讲解，并且针对图 1-13 所示的关系 R 进行讲解。

姓名	年龄	性别
李平	20	男
王梅	21	女
袁弘	20	男

图 1-13　关系 R

1. 选择(Selection)

假设 R 是 n 目关系，F 是命题公式，其结果为逻辑值，取“真”或“假”，则 R 的选择操作定义为：

$$\sigma_F(R)=\{t|t\in R\wedge F(t)=true\}$$

即取出满足条件 F 的所有元组。其中 F 包含下列两类符号：

运算对象有元组分量(属性名或列序号)、常数；运算符有＞、≥、＜、≤、＝、≠、¬、∧、∨。选择运算是从关系 R 中选取使逻辑表达式 F 为真的元组，是从行的角度进行的运算。

例如，对关系 R 进行以下查询的关系运算。

(1) 查询男生的信息。

$$\sigma_{性别='男'}(R)$$

(2) 查询年龄大于 20 的学生的信息。

$$\sigma_{年龄>20}(R)$$

(3) 查询年龄大于 20 的男学生的信息。

$$\sigma_{性别='男'\wedge 年龄>20}(R)$$

条件表达式 F 中的字符常量需要用单引号括起。选择操作是从关系里面选择满足条件 F 的元组，选择操作一般从行的角度进行筛选，有的数据库管理系统将选择操作称为水平筛选，选择操作的结果仍然是关系，结果的字段数量不会减少。

2. 投影(Projection)

投影操作是从关系 R 中选择出若干属性列组成新的关系。

$$\pi_A(R) = \{ t[A] \mid t \in R \}$$

A：关系 R 中的属性列。

投影操作主要是从列的角度进行运算，也就是选择关系的部分列而得到新的关系，投影又称为垂直筛选。投影操作之后不仅去掉了原关系中的某些字段，而且还可能取消某些元组(去掉重复的行)。

例如，在图 1-14 中查询关系 R_1 的年龄分布的关系代数为：

$$\pi_{年龄}(R_1)$$

姓名	年龄	性别
李平	20	男
王梅	21	女
袁弘	20	男

图 1-14　关系 R_1

得到的结果如下：

年龄
20
21

查询关系 R_1 的姓名和年龄的关系代数为：

$\pi_{姓名，年龄}(R_1)$

得到的结果如下：

姓名	年龄
李平	20
王梅	21
袁弘	20

3. 连接(Join)

连接分为内连接和外连接。内连接只将满足连接条件的元组保存在结果中；外连接除了将满足条件的元组保存在结果中，还把舍弃的元组也保存在结果关系中，并在其他属性上填空值(Null)。

外连接分为左外连接、右外连接和完全外连接。如果只把左边关系 R 中要舍弃的元组保留，就称为左外连接。如果只把右边关系 S 中要舍弃的元组保留，就称为右外连接。如果把左边关系和右边关系中不满足连接条件的元组也放在结果中，就称为完全外连接。

内连接也称为 θ 连接。连接运算的含义是从两个关系的笛卡儿积中选取属性间满足一定条件的元组。

内连接公式如下所示：

$$R \underset{A\theta B}{\infty} S=\{<t_r,t_s>|t_r \in R \wedge t_s \in S \wedge t_r[A]\theta t_s[B]\}$$

θ 运算符是比较运算符，如＞、＜、≠、=。A 和 B 分别是关系 R 和关系 S 上的一个属性或者多个属性组合。R 和 S 的连接运算是从 R 和 S 的广义笛卡儿积 R×S 中选取(R 关系)在 A 属性组上的值与(S 关系)在 B 属性组上的值满足比较关系 θ 的元组。

有两类最常用的连接运算：等值连接和自然连接。当连接符号 θ 为＝时的连接运算称为等值连接。等值连接的含义是从关系 R 与 S 的广义笛卡儿积中选取 A、B 属性值相等的那些元组而得到的关系。

自然连接是一种特殊的等值连接，等值连接中包含相同的字段，这样的关系看起来很不自然，为了让连接后的关系更加自然，两个连接关系中进行比较的字段必须是相同的属性或者属性组合，在结果中把重复的列去掉。

自然连接公式如下所示：

$$R \infty S=\{<t_r,t_s>|t_r \in R \wedge t_s \in S \wedge t_r[A]=t_s[B]\}$$

下面通过举例说明 θ 连接和自然运算，结果如下所示。

R 关系

A	B	C
a_1	b_1	3
a_2	b_1	5
a_3	b_2	5
a_4	b_3	6

S 关系

B	D
b_1	4
b_2	5
b_3	5
b_3	3

$R \underset{C<D}{\infty} S$

A	R.B	C	S.B	D
a_1	b_1	3	b_1	4
a_1	b_1	3	b_2	5
a_1	b_1	3	b_3	5

$R \underset{R.B=S.B}{\infty} S$

A	R.B	C	S.B	D
a_1	b_1	3	b_1	4
a_2	b_1	5	b_1	4
a_3	b_2	5	b_2	5
a_4	b_3	6	b_3	5
a_4	b_3	6	b_3	3

自然连接 $R \infty S$

A	B	C	D
a_1	b_1	3	4
a_2	b_1	5	4
a_3	b_2	5	5
a_4	b_3	6	5
a_4	b_3	6	3

1.5 数据库设计

如果一个数据库没有进行一个良好的设计，那么这个数据库完成之后存在以下缺点：效率会很低，更新和检索数据时会出现很多问题；反之，一个数据库被精心策划了一番，具有良好的设计，那么它的效率会很高，并便于进一步扩展，使应用程序的开发变得更容易。

数据库的设计步骤如下：

需求分析阶段：分析客户的业务和数据处理需求。

概要设计阶段：主要就是绘制数据库的 E-R 图。

详细设计阶段：应用数据库的三大范式审核数据库的结构。

1.5.1 实体联系图(E-R 图)

E-R 图也称实体联系图(Entity Relationship Diagram)，提供了表示实体类型、属性和联系的方法，用来描述现实世界的概念模型。每一类数据对象的个体称为实体，而每一类对象个体的集合称为实体集，如学生是一个实体集，张三是一个实体，姓名是一个属性。

两个实体之间的联系包括一对一(1∶1)、一对多(1∶n)和多对多(m∶n)三种。比如，一个学校只能有一个校长，而一个校长也只能担任一个学校的校长。学校和校长之间的联系就是一对一联系。一个学校里有多名教师，而每个教师只能在一个学校教学，学校和教师之间的联系就是一对多联系。一个学生可以上 n 门课程，而每一门课程可以有 m 个学生学习。课程和学生实体之间的联系就是多对多联系。联系可以有自己的属性，如学生和课程之间有选课联系每个选课联系都有一个成绩作为其属性，成绩属性描述某个学生选修某门课程的成绩。

E-R 图的四个组成部分如下。

(1) 矩形框：表示实体，在矩形框中写上实体的名称。

(2) 椭圆形框：表示实体或联系的属性。

(3) 菱形框：表示联系，在框中写上联系名。

(4) 连线：实体与属性之间、实体与联系之间、联系与属性之间用直线相连。对于一对一联系，要在两个实体连线方向各写 1；对于一对多联系，要在一的一方写 1，多的一方写 n；对于多对多关系，则要在两个实体连线方向各写 n、m。

1.5.2 规范化理论

关系模型有严格的数学理论基础，因此人们就以关系模型作为讨论对象，形成了数据库逻辑设计的一个有力工具——关系数据库的规范化理论。关系数据库的规范化设计是指面对一个现实问题，如何选择一个比较好的关系模式集合。规范化设计理论对关系数据库结构的设计起着重要的作用。

什么是好的数据库呢？我们在设计关系模式时，能不能将所有的信息放在一张表里面？构建好的、合适的数据库模式是数据库设计的基本问题，如果数据库没有进行相应的规范设计，虽然在查询数据库时可能会比较容易，但有时会造成一些问题，主要有以下几个问题。

- 信息重复(会造成存储空间的浪费及一些其他问题)。
- 更新异常(冗余信息不仅浪费空间，还会增加更新的难度)。

- 插入异常。
- 删除异常(在某些情况下，当删除一行时，可能会丢失有用的信息)。

好的数据库设计，体现客观世界的信息，而且无过度冗余、无插入异常、无删除异常、无更新复杂。

假设需要设计一个学生学习情况数据库。下面我们以模式 SCG(学号，姓名，年龄，所在系，课程号，课程名，学分，成绩)为例来说明将所有信息都放在这张表里面存在的问题。

- 冗余度大：每选一门课，他本人信息和有关课程信息都要重复一次。
- 插入异常：插入一门课，若没学生选修，则不能把该课程插入表中。
- 删除异常：如 S11 号学生的删除，有一门只有他选，会造成课程的丢失。
- 更新复杂：更新一个人的信息，则要同时更新很多条记录。还有更新选修课时也存在这样的情况。

异常的原因是数据存在依赖约束。解决方法是数据库设计的规范化：分解，每个相对的独立，依赖关系比较单纯，如分解为第 3 范式(3NF)。

可以采用分解的方法，将上述 SCG 分解成以下三个模式(也就是一个表分为三个表)：

S(学号，姓名，年龄，所在系)

C(课程号，课程名，学分)

SC(学号，课程号，成绩)

函数依赖(Functional Dependency，FD)是指一个或一组属性可以(唯一)决定其他属性的值。

数学的语言：

设有关系模式 R(U)，其中 U={A_1，A_2，…，A_n}是关系的属性全集，X、Y 是 U 的属性子集，设 t 和 u 是关系 R 上的任意两个元组，如果 t 和 u 在 X 的投影 t[X]=u[X]推出 t[Y]=u[Y]，即 t[X]＝u[X] => t[Y]＝u[Y]，则称 X 函数决定 Y，或 Y 函数依赖于 X，记为 X→Y。在上述关系模式 S(学号，姓名，年龄，所在系)中，存在以下函数依赖：

学号→年龄

学号→姓名

(学号，课程号)→成绩

完全函数依赖和部分函数依赖：设 X、Y 是关系模式 R 的不同属性集，若 X→Y (Y 函数依赖于 X)，并且对于 X 的任意一个真子集 X′都有 X′→Y，则称 Y 完全函数依赖于 X(即不存在真子集仍然是函数依赖关系的函数依赖是完全函数依赖)，否则称 Y 部分函数依赖于 X。

例如，在上例关系模式 S 中，姓名是完全依赖于学号；成绩是部分依赖于学号。

在属性 Y 与 X 之间，除了存在完全函数依赖和部分函数依赖等直接函数依赖关系外，还存在间接函数依赖关系。如果在关系模式 S 中增加系的办公电话字段，从而有学号→系名，系名→办公电话，于是有学号→办公电话。在这个函数依赖中，办公电话并不直接依赖于学号，是通过中间属性系名间接依赖于学号，这就是传递函数依赖。

1.5.3 关系模式的规范化

1. 什么是范式(Normal Forms)

构造数据库必须遵循一定的规则，满足特定规则的模式称为范式。一个关系满足某个范式

所规定的一系列条件时，它就属于该范式。可以用规范化要求来设计数据库,也可验证设计结果的合理性，用其来指导优化数据库设计过程。

关系规范化条件可分为几级，每级称为一个范式，记为第 x 范式。

1NF→2NF→3NF→BCNF，4NF→5NF

级别越高，条件越严格，高级的范式包含低级的范式，例如，一个关系模式满足第 2 范式，则一定满足第 1 范式。

范式是衡量模式优劣的标准，范式表达了模式中数据依赖之间应满足的联系。如果关系模式 R 是 3NF，那么 R 上成立的非平凡 FD 都应该左边是超键或右边是非主属性。如果关系模式 R 是 BCNF，那么 R 上成立的非平凡的 FD 都应该左边是超键。范式的级别越高，其数据冗余和操作异常现象就越少。

2. 第 1 范式(1NF)

如果一个关系模式 R 的每个属性的域都只包含单纯值，而不是一些值的集合或元组，则称关系是第 1 范式，记为 R∈1NF。

或者，如果关系模式 R 的每个关系 r 的属性值都是不可分的原子值，那么称 R 是第一范式(理解：每个元组的每个属性只含有一个单纯值，即要求属性是原子的)。这是关系模式的基本要求，条件是最松的，只要你不硬把两个属性塞到一个字段中去。如果不满足 1NF，就不是关系数据库。比如下表是不满足第一范式的关系模式。

字段 1	字段 2	字段 3		字段 4
		属性 1	属性 2	

把一个非规范化的模式变为 1NF 有以下两种方法。

(1) 把不含单纯值的属性分解为多个属性，使它们仅含有单纯值。

例：通信方式分为电话、手机、邮编、地址等。

例：Name(First Name，Last Name)

(2) 把关系模式分解，并使每个关系都符合 1NF。

下面介绍列和行的原子属性。

列的原子属性：每个字段不再分割成多个属性。

行的原子属性：每个元组在表中只可出现一次。

第一范式中一般情况下都会存在数据的冗余和异常现象，因此关系模式需要进行进一步的规范化。

3. 第 2 范式(2NF)

它是在 1NF 的基础上建立起来的。如果关系模式 R∈1NF，且它的任一非主属性都完全函数依赖于任一候选关键字，则称 R 满足第 2 范式，记为 R∈2NF。(理解：不存在非主属性对关键字的部分函数依赖)。

例：学生(学号，课程号，成绩，学分)就不满足第二范式，因为学分是部分依赖于主属性。因此，学生∈1NF。

例：S(学号，姓名，年龄，系名，办公电话)，因为每个非主属性对关键字 S 都是完全函数

依赖的，S∈2NF。由上例可知，2NF 依然有较多冗余(办公电话)，继续分解，提高条件。

4. 第 3 范式(3NF)

如果 R∈2NF，且每一个非主属性不传递依赖于任一候选关键字，则称 R∈3NF。(理解：任一属性不依赖于其他非主属性)。

例：S(学号，姓名，年龄，系名，办公电话)中办公电话属性对关键字学号是传递函数依赖的，因此关系 S 不满足第 3 范式。

通过分解：

S(学号，姓名，年龄，系名，办公电话)

D (系名，办公电话)

上述两个关系就满足第 3 范式了，每个非主属性既不部分依赖也不传递依赖于候选关键字。

1.6 小结

数据库是大量结构化的数据的集合，数据管理发展经历了三个阶段：人工管理阶段、文件系统阶段和数据库系统阶段。E-R 模式是数据库设计人员之间交流的工具，实体之间的联系包括一对一、一对多和多对多的联系，规范化理论是设计数据库的理论基础，通过规范化可消除关系的异常。

1.7 练习题

选择题

1. 支持数据库各种操作的软件系统称为(　　)。

A. 命令系统　　B. 数据库系统　　C. 操作系统　　D. 数据库管理系统

2. 在 Access 数据库中，与关系模型中“域”对应的概念是(　　)。

A. 字段的取值范围　　B. 字段的默认值

C. 字段的数据类型　　D. 字段的显示格式

3. 在实体关系模型中，有关系 R(学号，姓名)、关系 S(学号，课程编号)和关系 P(课程编号，课程名)要得到关系 Q(学号，姓名，课程名)，应该使用的关系运算是(　　)。

A. 连接　　B. 选择　　C. 投影　　D. 无法实现

4. Access 中，与关系模型中的概念“元组”相对应的术语是(　　)。

A. 字段　　B. 记录　　C. 表　　D. 域

5. 如果“主表 A 与相关表 B 之间是一对一联系”，它的含义是(　　)。

A. 主表 A 和相关表 B 均只能各有一个主关键字字段

B. 主表 A 和相关表 B 均只能各有一个索引字段

C. 主表 A 中的一条记录只能与相关表 B 中的一条记录关联

D. 主表 A 中的一条记录只能与相关表 B 中的一条记录关联，反之亦然

6. 要在表中检索出属于计算机学院的学生，应该使用的关系运算是(　　)。

A. 连接　　B. 关系　　C. 选择　　D. 投影

7. 在关系数据库中，关系是指(　　)。

A. 各条记录之间有一定的关系　　B. 各个字段之间有一定的关系

C. 各个表之间有一定的关系　　D. 满足一定条件的二维表

8. 下列与 Access 表相关的叙述中，错误的是(　　)。

A. 设计表的主要工作是设计表的字段和属性

B. Access 数据库中的表由字段和记录构成

C. Access 不允许在同一个表中有相同的数据

D. Access 中的数据表既相对独立又相互联系

9. 在 Access 2016 中，对数据库对象进行组织和管理的工具是(　　)。

A. 工作区　　B. 导航窗格　　C. 命令选项卡　　D. 数据库工具

10. Access 中存储基本数据的对象是(　　)。

A. 表　　B. 查询　　C. 窗体　　D. 报表

11. 使用 Access 数据库管理技术处理的数据不仅可以存储为数据库文件，还可以多种文件格式导出数据，以下不支持导出的文件格式是(　　)。

A. Word 文件　　B. Excel 文件　　C. PDF 文件　　D. PNG 文件

12. 若有关系(课程编号，课程名称，学号，姓名，成绩)，要得到关系中有多少门不同的课程名称，应使用的关系运算是(　　)。

A. 连接　　B. 关系　　C. 选择　　D. 投影

13. 下列关于关系模型特点的叙述中，错误的是(　　)。

A. 一个数据库文件对应着一个实际的关系模型

B. 一个具体的关系模型由若干关系模式所组成

C. 在一个关系中属性和元组的次序都是无关紧要的

D. 可将手工管理的表按一个关系直接存到数据库中

14. 一个元组对应表中的(　　)。

A. 一个字段　　B. 一个域　　C. 一个记录　　D. 多个记录

15. 在关系数据模型中，域是指(　　)。

A. 字段　　B. 记录　　C. 属性　　D. 属性的取值范围

16. 下列关于数据库的叙述中，正确的是(　　)。

A. 数据库避免了数据的冗余

B. 数据库中的数据独立性强

C. 数据库中的数据一致性是指数据类型一致

D. 数据库系统比文件系统能够管理更多数据