# 第 1 章　数据库基础知识

数据库技术是数据管理的技术，是计算机科学与技术的重要分支，是信息系统的核心和基础。当今社会上各种各样的信息系统都是以数据库为基础，对信息进行处理和应用的系统。数据库能借助计算机保存和管理大量的、复杂的数据，快速而有效地为不同的用户和各种应用程序提供重要的数据，以便人们能更方便、更充分地利用这些数据。

数据库技术的发展，已经成为先进信息技术的重要组成部分，是现代计算机信息系统和计算机应用系统的基础与核心。数据库技术从 20 世纪 60 年代后期产生到今天仅仅几十年的历史，但已经得到了迅速发展和广泛应用，成为计算机科学与技术的一个重要分支。

在本章中，主要介绍数据处理技术的发展、数据库的几个重要概念、数据模型、关系模型和关系数据库，并描述关系数据库的设计步骤，为后面各章的学习打下基础。

## 1.1　数据处理技术

### 1.1.1　信息、数据与数据处理

#### 1. 信息与数据

信息是客观事物属性的反映。它所反映的是某客观系统中某一事物的存在方式或某一时刻的运动状态。通俗地讲，信息是经过加工处理并对人类客观行为产生影响的、通过各种方式传播、可被感知的数据表现形式。

数据是反映客观事物属性的记录，是信息的载体。对客观事物属性的记录是通过一定的符号来表达的，因此说数据是信息的具体表现形式。数据表现信息的形式是多种多样的，不仅包括数字、文字组成的文本形式，而且还包括图形、图像、动画和声音等多媒体形式。用数据记录同一信息可以有不同的形式，信息不会随着数据形式的不同而改变其内容和价值。

从计算机角度来说，数据泛指那些可以被计算机接受并能够被计算机处理的符号，是数据库中存储的基本对象。

信息与数据在概念上是有区别的。从信息处理角度看，任何事物的存在方式或运动状态都可以通过数据来表示，数据经过加工处理后，具有知识性并对人类的活动产生影响，从而形成信息。

总之，信息是有用的数据，数据是信息的表现形式。信息通过数据符号传播，数据如不具有知识性和有用性则不能称其为信息。

数据和信息的关系非常密切，在某些不需要严格区分的场合，可以将二者不加区别地使用。如将信息处理说成是数据处理。

**2. 数据处理**

数据处理也称为信息处理。所谓数据处理，是指将数据加工成信息的过程，它包括对数据的收集、整理、存储、分类、排序、检索、维护、加工、统计和传输等一系列操作过程。数据处理的目的是从收集的大量原始数据中获得所需要的资料并提取有用的数据成分，作为行为和决策的依据。数据处理的核心是数据管理。

### 1.1.2 数据处理技术的发展

早期的计算机主要用于科学计算。当计算机应用于财务管理、图书资料管理、仓库管理等领域时，它所面对的是大量的各种类型的数据。为了有效地管理和利用这些数据，就产生了计算机的数据管理技术。

随着计算机软硬件技术的发展，数据处理量的规模日益扩大，数据处理的应用需求越来越广泛，数据处理技术的发展也不断变迁，经历了人工管理、文件管理和数据库系统这3个发展阶段。

**1. 人工管理阶段**

20世纪50年代中期以前，计算机主要用于数值计算。在这一阶段，只有卡片、纸带、磁带能用于存储数据，软件方面还没有操作系统，没有进行数据管理的软件。

在人工管理阶段，应用程序与数据之间的关系如图1.1所示。

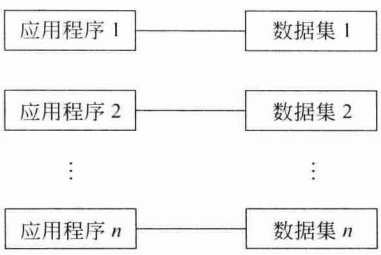


图 1.1 人工管理阶段应用程序与数据的关系

在这一阶段，数据管理的特点如下。

(1) 数据不保存。程序员将程序和数据编写在一起输入内存，程序对数据进行处理后输出处理结果。程序运行结束后，数据也将从内存释放。

(2) 应用程序与数据之间缺乏独立性。应用程序与数据之间相互依存，不可分割；编写程序时要安排数据的物理存储，当数据有所变动时，应用程序也随之变动。程序员的工作量大、烦琐，程序难于维护。

(3) 数据无法共享，数据重复存储，冗余度大。

**2. 文件管理阶段**

20世纪60年代中后期，硬件方面出现了磁带、磁盘等大容量存储设备，软件方面出现了操作系统。在这一阶段，由于使用专门的操作系统中的文件管理系统实施数据管理，数据被组织成数据文件，可以脱离应用程序而独立存在。

用户的应用程序与数据文件可分别存放在外存储器上，不同的应用程序可以共享一组数据，实现了数据以文件为单位的共享，如图1.2所示。

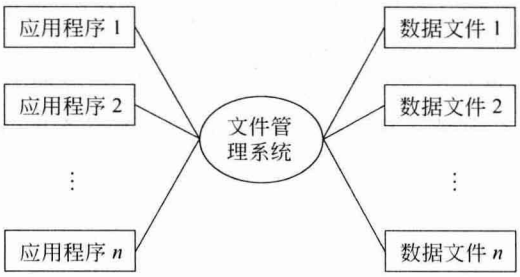


图 1.2　文件管理阶段应用程序与数据的关系

在文件管理阶段数据处理的特点如下。

(1) 数据可以长期保存。

(2) 应用程序与数据之间有了一定的独立性。

(3) 数据文件有了一定的共享性，但仍存在较大的数据冗余，数据还未达到完全的一致性。

**3. 数据库系统阶段**

进入 20 世纪 60 年代后期，随着计算机应用领域的日益发展，计算机在数据处理方面的应用越来越广泛，处理的数据量越来越大，仅仅基于文件系统的数据处理技术很难满足应用领域的需求；与此同时，出现了大容量且价格低廉的磁盘，改善数据处理软件的功能成为许多软件公司的重要目标。在实际需求迫切，硬件与软件技术发展趋于成熟的条件下，出现了数据库技术和统一管理数据的专门软件系统——数据库管理系统。1968 年，美国 IBM 公司研制成功的信息管理系统(Information Management System，IMS)标志着数据处理技术进入了数据库系统阶段。

数据库系统对相关数据实行统一规划管理，形成一个数据中心，构成一个数据仓库，实现了整体数据的结构化。用数据库系统管理数据比文件系统有明显的优势，从文件系统到数据库系统，标志着数据管理技术的飞跃。

在数据库管理系统的支持下，应用程序与数据之间的关系如图 1.3 所示。在这一阶段，系统可以有效地管理和存取大量的数据，提高了数据的共享性，使多个用户可以同时访问数据库中的数据，减少了数据冗余，保证了数据的一致性和完备性，数据与应用程序相对独立，减少了应用程序开发和维护的成本。

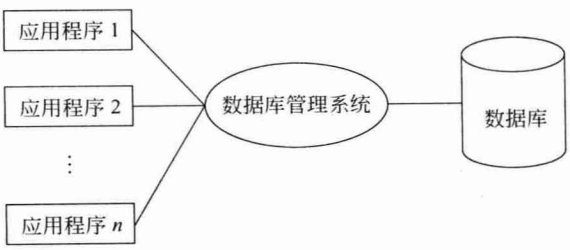


图 1.3　数据库系统中应用程序与数据的关系

20 世纪 80 年代中后期，计算机技术不断应用到各行各业，数据库应用领域不断扩展，用户的需求呈现多样化和复杂化，需要存储的数据量也越来越庞大，数据之间的结构越来越复杂。因此，传统的关系型数据库已经不能完全满足需求，由此产生了新一代数据库技术。

新一代数据库技术与其他学科相结合，涌现出了各种新型的数据库系统。例如，分布式数据库系统、并行处理数据库系统、知识库系统、主动数据库系统、多媒体数据库系统和模糊数据库系统等。

(1) 分布式数据库系统。分布式数据库允许用户开发的应用程序把多个物理分开的、通过网络互联的数据库当作一个完整的数据库看待。用户不必关心数据的分片，不必关心数据物理位置分布的细节，不必关心数据副本的一致性，分布的实现完全由分布式数据库系统来完成。

(2) 并行处理数据库系统。并行处理数据库系统通过将数据库在多个磁盘上分布存储，可以利用多个处理机对磁盘数据进行并行处理，解决了磁盘 I/O 瓶颈问题，通过开发查询的并行性、查询内的并行性以及操作内的并行性，提高了数据库的吞吐率、容错性和查询的效率。

(3) 知识库系统。知识库系统是数据库和人工智能两种技术结合的产物。使传统数据库具有一些人工智能的能力(如专家系统)，以提高数据库的演绎、推理功能和智能化的程度，发展智能化的数据库。

(4) 主动数据库系统。主动数据库系统通常在传统数据库系统中嵌入 EAC(即事件—条件—动作)规则，在某一事件发生时引发数据库管理系统去检测数据库当前状态，如满足设定的条件，便触发规定动作的执行。主动数据库系统提供对紧急情况及时反应的能力，同时提高数据库管理系统的模块化程度。

(5) 多媒体数据库系统。多媒体数据库提供了一系列用来存储图像、音频和视频对象的类型，更好地对多媒体数据进行存储、管理和查询。

(6) 模糊数据库系统。模糊数据库是存储、组织、管理和操作模糊数据的数据库，可以用于模糊知识处理。目前，模糊数据库系统还不够完善，但是已在模式识别、过程控制、案件侦破、医疗诊断、专家系统等领域有较好的应用。

### 1.1.3 数据库系统

#### 1. 数据库系统的组成

数据库系统(DataBase System，DBS)是引入了数据库的计算机系统，它包含计算机的软硬件系统、数据库、数据库管理系统、数据库应用系统、数据库管理员和一般用户，如图 1.4 所示。

(1) 数据库。所谓数据库(Data Base，DB)，是以一定的组织方式将相关的数据组织在一起，长期存放在计算机内，可为多个用户共享，与应用程序彼此独立，统一管理的数据集合。

数据库中的数据按一定的数据模型描述、组织和存储，具有较小的冗余度，较高的数据独立性和易扩展性，它不仅要反映数据本身的内容，而且要反映数据之间的关系，并可为一定范围内的各种用户共享。

(2) 数据库管理系统。数据库管理系统(Data Base Management System，DBMS)是

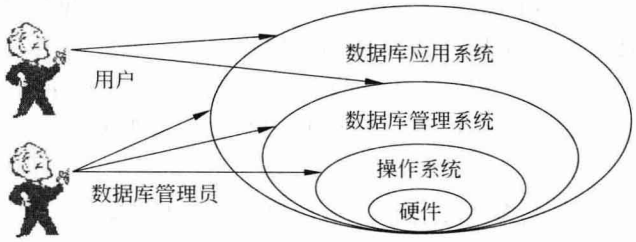


图 1.4　数据库系统的组成

实现对数据库进行管理的一系列软件的集合，它以统一的方式管理和维护数据库，并提供数据库接口供用户访问数据库。主要功能有：

① 定义数据库。数据库管理系统提供了定义数据类型以及数据库存储形式的功能。用户可根据需要在计算机中创建数据库和定义数据库结构，并存储用户输入的数据。

② 操作数据库。数据库管理系统提供了多种处理数据的操作方式。用户可根据需要对数据库进行添加、修改、删除和检索等操作。

③ 管理和维护数据库。数据库管理系统提供了对数据进行维护和管理的功能，在多用户同时对数据库进行访问时，能保证数据的安全性和完整性，还提供初始数据的导入、管理过程中数据的存储、数据的备份等，保证数据访问的正确无误，实现故障处理和性能监视，达到对大量数据的管理及维护功能。

数据库管理系统是数据库系统的核心，其主要工作就是管理数据库，为用户或应用程序提供访问数据库的方法。

(3) 数据库应用系统。数据库应用系统(DataBase Application System，DBAS)是指软件开发人员利用数据库管理系统提供的功能，对数据库进行管理和应用而开发的，方便用户使用的，应用于某一个实际问题的应用软件。如学生成绩管理系统、学籍管理系统、人事档案管理系统、图书借阅系统、用于大型企业的信息管理系统等都属于数据库应用系统。

(4) 人员。数据库系统的人员是指管理和使用数据库系统的全部人员，主要包括数据库管理员和一般用户。

① 一般用户主要通过数据库应用系统提供的用户界面使用数据库，是数据库的使用者。

② 数据库管理员(DataBase Administrator，DBA)负责技术层的全局控制，主要有以下 3 方面的具体工作。

- 数据库设计。对数据的需求做全面的规划、设计和集成，这是数据库管理员的基本任务。
- 数据库维护。对数据库中数据的安全性、完整性、并发控制及系统恢复进行实施与维护。
- 改善系统性能和提高系统效率。随时监视数据库的运行状态，不断调整内部结构，保持系统的最佳状态与最高效率。

**2. 数据库系统的特点**

(1) 数据的结构化。这是数据库系统与文件系统的根本区别。数据库系统中的数据

是有结构的，这些数据由数据库管理系统进行统一的管理。在数据库系统中，数据不再针对某一个应用，而是面向全组织，形成整体的结构化。

(2) 数据的共享程度高、易扩充、冗余度低。数据库系统从整体规划角度来描述系统中存储的数据，数据由数据库系统统一管理、集中存储。数据面向整个系统应用，而且容易增加新的应用，易于扩充。

由于数据的集成性使得数据可为多个应用所共享，特别是发达的网络扩大了数据库的应用范围。数据的共享性高又减少了数据的冗余度，避免了数据的不一致性。

数据冗余是指数据库中数据的重复存储，数据冗余不仅浪费了大量的存储空间，而且还会影响数据的正确性和一致性。数据冗余是不可避免的，但是由于数据库内的数据可以被多个用户、多个应用共享，所以可最大限度地减少数据的冗余度。

(3) 数据的独立性高。数据的独立性包括数据的物理独立性和逻辑独立性。

数据的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。它的优点是数据的物理存储结构即使改变了，用户的应用程序也不需要跟着改变。

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。它的优点是即使数据的逻辑结构改变了，用户的应用程序也可以保持不变。

(4) 数据控制功能较强。数据库中的数据被多个用户或应用程序所共享。当多个用户同时存取或修改数据库中的数据时，对于相互之间发生的干扰，产生的错误数据，甚至破坏数据库等不良行为，数据库管理系统都能提供较强的保护控制功能，包括数据的并发控制、数据的安全性控制和数据的完整性控制，避免错误数据的产生。

## 1.2 数据模型

数据库是某个企业、公司或部门所涉及数据的集合，它不仅要反映数据自身，而且要反映数据之间的联系。数据模型是对现实世界特征的模拟和抽象。由于计算机不可能直接处理现实世界中的具体事物，所以人们需要事先把具体事物转换成计算机能够处理的数据，在数据库中采用数据模型来抽象、表示和处理现实世界中的数据和信息。

数据模型是数据库中数据的存储方式，是数据库系统的核心和基础，它描述了不同数据之间的关系，决定了数据库的设计方法。为了将现实世界中的具体事物抽象，组织成数据库管理系统支持的数据模型，需要有以下两个步骤。

(1) 将现实世界抽象为信息世界，创建概念模型来描述数据。

(2) 将信息世界转换为计算机世界，用计算机能接受的数据模型(包括层次模型、网络模型和关系模型)来描述数据。

图 1.5 描述了从现实世界到计算机世界的抽象过程。

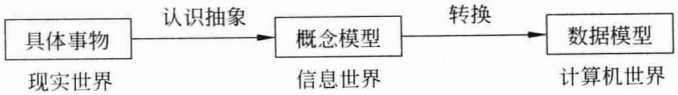


图 1.5 现实世界到计算机世界的抽象过程

### 1.2.1　概念模型

概念模型是一种面向客观世界、面向用户的模型，它与具体的数据库管理系统和具体的计算机平台无关。这类模型概念简单、清晰，易于被用户理解，是用户和数据库设计人员之间进行交流的语言。最著名的概念模型是实体-联系模型（Entity-Relationship Model，E-R 模型），目前在数据库设计中广泛使用此模型。

E-R 模型采用了 3 个基本概念：实体、属性和联系。

**1. 实体**

实体是客观事物的真实反映，既可以是实际存在的对象，如一名教师、一名学生、一本书等，又可以是某种抽象概念或事件，如一门课程、一次授课、一个班级、成绩表、一次借阅图书等。

**2. 属性**

实体所具有的某一方面的特性，称为**属性**。每个实体通常都具有多个属性，如学生实体有学号、姓名、性别、出生日期、政治面貌等多个属性，教师实体有教师编号、姓名、性别、工作时间、职称等多个属性。属性由属性名和属性值两部分构成，一个属性的取值范围称为该属性的**值域**。如性别只能从“男”或“女”两个值中选择，所以性别的值域为{男，女}；成绩的值域为[0，100]。

实体的某一属性或属性的组合，它的值能唯一标识出某一个实体，称为**关键字**，也称为**码**。如“学号”属性是学生实体的关键字，“学号”属性和“课程号”两个属性共同作为成绩表实体的关键字。

在本书以后的描述中，关键字用下划线标识。

**3. 实体型和实体集**

具有相同属性的实体具有相同的特性，用实体名和属性名集合来抽象和刻画同类实体，称为**实体型**。如学生（学号，姓名，性别，出生日期，政治面貌，班级）就是一个实体型。

同类实体的集合称为实体集。如全体学生就是一个实体集。

**4. 联系**

现实世界是一个有机的相互关联的整体，这种关联在概念模型中表现为实体之间的对应关系。通常将实体之间的对应关系称为联系。实体之间的联系有一对一、一对多和多对多三类。

（1）一对一联系（1∶1）。一对一联系是指一个实体和另一个实体之间存在着一一对应关系。如一所学校只有一名校长，并且一名校长只能在一所学校任职，不可以在别的学校任职，校长与学校之间的联系就是一对一的联系，如图 1.6(a)所示。同样，班级与班长之间、行进中的汽车与司机之间的联系都是一对一的联系。

（2）一对多联系（1∶$n$）。一对多联系是指一个实体对应着多个实体。如一个班级有多名学生，并且一名学生只能属于一个班级，班级和学生之间的联系就是一对多的联系，如图 1.6(b)所示。同样，系与教师之间、企业与职工之间的联系都是一对多的联系。

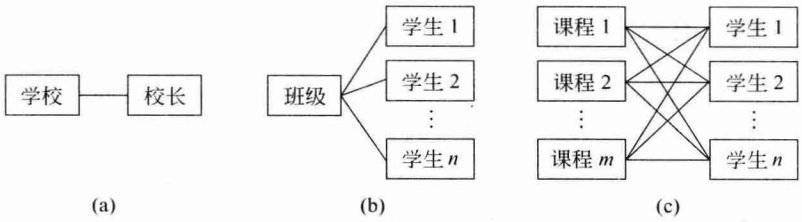


图 1.6 联系类型示意图

(3) 多对多联系($m:n$)。多对多联系是指多个实体对应着多个实体。如一名学生选修多门课程,并且一门课程有多名学生选修,课程和学生之间的联系就是多对多的联系,如图 1.6(c)所示。同样,供应商与商品之间、教师与课程之间、教师与学生之间的联系都是多对多的联系。

在现实生活中,多对多联系是普遍现象,一对多、一对一联系仅是多对多联系的特例。

**5. E-R 图**

E-R 图用图形的方式描述概念模型,它通用的表现方式如下。

(1) 用长方形表示实体,在框内写上实体名。

(2) 用椭圆形表示属性,并用直线把实体与属性连接起来。

(3) 用菱形表示实体间的联系,菱形框内写上联系名。用直线把菱形与相关实体连接,在直线上标上联系的类型。如果联系有自己的属性,则把属性和菱形也用直线连接起来。

图 1.7 描述的是课程和学生之间的 E-R 图。

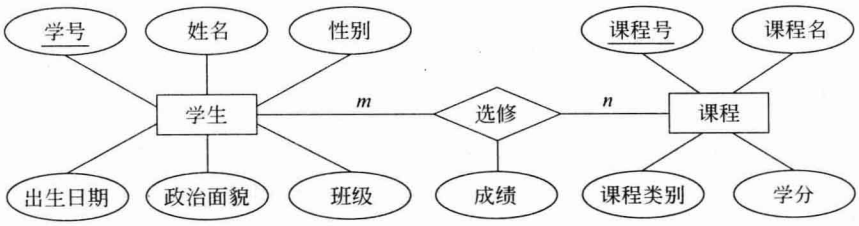


图 1.7 课程和学生的 E-R 图

用 E-R 图表示的概念模型与具体的数据库管理系统所支持的数据模型相独立,是各种数据模型的共同基础,因而比数据模型更一般、更抽象、更接近现实世界。

## 1.2.2 数据模型

从数据库的逻辑结构出发,对数据库中的实体、实体之间的联系进行描述,构成了数据模型。在几十年的数据库发展史中,出现了 4 种重要的数据模型:层次模型、网络模型、关系模型和面向对象模型。

**1. 层次模型**

层次模型是数据库系统中最早采用的数据模型,它用树状结构组织数据。

在树状结构中，各个实体被表示为结点，结点之间具有层次关系。相邻两层结点称为父子结点，父结点与子结点之间构成了一对多的关系。

有且仅有一个根结点（无父结点），其余结点有且仅有一个父结点，但可以有零个或多个子结点。

它的优点是简单、直观且处理方便，适合表现具有比较规范的层次关系的结构，在现实世界中存在着大量可以用层次结构表示的实体，如单位的行政组织结构、家族关系、磁盘上的文件夹结构等都是典型的层次结构，图 1.8 描述了某个高校的行政组织结构。

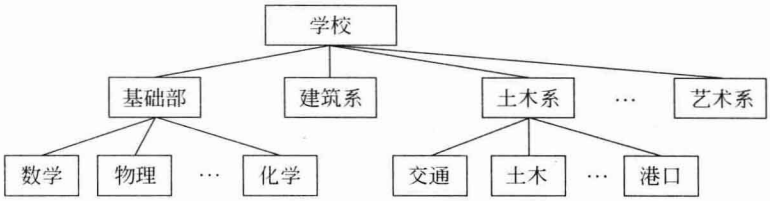


图 1.8　层次模型示例

**2. 网状模型**

网状模型是层次模型的扩展，用图的方式表示数据之间的关系。网状模型可以方便地表示实体间多对多的联系，但结构比较复杂，数据处理比较困难，如公交线路中各个站点之间的关系、城市交通图等都可以用网状模型来描述。图 1.9 描述了教学管理中各实体之间的联系。

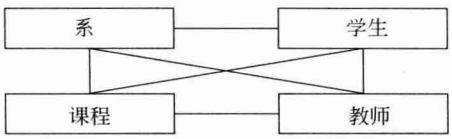


图 1.9　网状模型示例

**3. 关系模型**

关系模型是用二维表表示实体与实体之间联系的模型，它的理论基础是关系代数。关系模型中数据以表的形式出现，操作的对象和结果都是二维表，每一个二维表称为一个关系，它不仅描述实体本身，而且还能反映实体之间的联系。

在上面介绍的三种数据模型中，层次模型和网状模型由于其使用的局限性，现在已经很少用了，目前应用最广泛的是关系模型。

**4. 面向对象模型**

面向对象模型是面向对象概念与数据库技术相结合的产物，该模型吸收了层次、网状和关系模型的优点并借鉴面向对象的设计方法，可以描述上述 3 种模型难以处理的复杂数据结构，具有较好的灵活性、可重用性及可扩充性。但由于该模型相对比较复杂，涉及的知识比较多，因此尚未达到普及的程度。

### 1.2.3　关系模型

关系模型具有坚实的数学理论基础，从 20 世纪 80 年代以来，软件开发商提供的数据库管理系统几乎都是支持关系模型的。目前常用的数据库管理系统 Visual FoxPro、

Oracle、SQL Server、Access、MySQL 等都是关系数据库。

**1. 关系模型的基本术语**

(1) 关系。关系是用二维表的结构表示实体和实体之间的联系，它由行和列组成。一个关系对应一张二维表。例如表 1.1 和表 1.2 的两张表对应两个关系。

**表 1.1 学生关系**

属性

| 学号 | 姓名 | 性别 | 出生日期 | 政治面貌 | 班级 |
|---|---|---|---|---|---|
| 09220101 | 张三 | 男 | 1990-12-21 | 群众 | 09 土木 1 |
| 09220102 | 郭永庆 | 男 | 1900-3-12 | 团员 | 09 土木 1 |
| 09220103 | 吕元昭 | 女 | 1990-3-14 | 团员 | 09 土木 1 |
| 09220104 | 唐阳 | 女 | 1990-9-19 | 团员 | 09 土木 1 |
| 09220105 | 张春本 | 男 | 1990-7-21 | 党员 | 09 土木 1 |
| 09220106 | 高文龙 | 男 | 1990-5-13 | 团员 | 09 土木 1 |

元组 —— 09220101

分量 —— 09220103 —— 吕元昭

**表 1.2 成绩表关系**

| 学 号 | 课 程 名 | 成绩 | 学 号 | 课 程 名 | 成绩 |
|---|---|---|---|---|---|
| 09220101 | 大学计算机基础 | 87 | 09220104 | 高等数学 | 76 |
| 09220102 | 高等数学 | 98 | 09220105 | 英语 | 85 |
| 09220103 | 大学计算机基础 | 78 | 09220106 | 英语 | 80 |

(2) 元组。在二维表中，每一行称为一个元组，它存储一个具体实体的信息。如学生关系有 6 行，因此它有 6 个元组。

(3) 属性。表中的一列称为一个属性。如学生关系共 6 列，因此它有学号、姓名、性别等 6 个属性。

(4) 分量。每个元组的一个属性值。如学生关系的第 1 个元组的姓名分量为“张三”。

(5) 域。属性的取值范围，即不同元组对同一个属性的取值所限定的范围。如学生关系的“性别”属性域为{男，女}。

(6) 候选码(关键字)。在关系中可以用来唯一地标识或区分一个元组的属性或属性组称为候选码。如学生关系中的“学号”是候选码，它可以唯一地确定每一个元组；而在成绩表关系中，“学号”不能单独成为候选码，属性组{学号，课程名}可以唯一确定一个学生的某门课程的成绩，所以共同组成候选码。

在一个关系中，候选码可以有多个。例如有零件关系(零件号，零件名称，产地，单价，数量，仓库号，存放位置)，“零件号”、{零件名称，产地}都能唯一地确定这个零件，所以都是零件关系的候选码。

(7) 主码(主关键字)。一个关系中可能有多个候选码，但在实际应用中只能选择一个使用，被选用的候选码称为主码。例如零件关系的主码我们选择为“零件号”。

(8) 外码。如果关系 R 中某个属性或属性组是关系 S 的主码，那么该属性或属性组是关系 R 的**外码**，关系 S 称为该外码的参照关系。如上例中成绩表关系中的学号是外码(学号是学生关系的主码)。

**2. 关系的特点**

(1) 关系中每一个分量不可再分，是最基本的数据单位。

(2) 每一列的分量是同属性的，各列的顺序可以任意交换，交换后不影响数据的使用。

(3) 每一行由实体的多个属性分量组成，各行的顺序可以任意交换，交换后不影响数据的使用。

(4) 一个关系是一张二维表，不允许有相同的属性名，也不允许有完全相同的两个元组。

**3. 关系的完整性约束**

关系的完整性约束是指关系中的数据及具有关联关系的数据间必须遵循的约束关系，关系的完整性用于保证数据的正确性和有效性。

关系模型提供了 3 种完整性约束。

(1) 实体完整性约束。关系的实体完整性约束是对关系中元组的唯一性约束，而关系中的主码可以标识每个元组，所以关系的实体完整性约束实质上是对关系中主码的约束，要求主码不能有空值，不能有重复值。

在关系数据库管理系统中，一个关系只能有一个主码，系统会自动进行实体完整性检查。

(2) 用户自定义完整性约束。用户自定义完整性约束是针对某个具体数据库，由用户自定义的约束条件。其作用是将某些属性的值限制在合理的范围内，对于超出正常值范围的数据系统将报警，同时这些非法数据无法进入数据库中。如“性别”属性的取值只能是“男”或“女”，“学分”属性的取值范围只能是[1,5]等。

(3) 参照完整性约束。参照完整性约束是对相关联的两个或多个关系之间的约束，它是由外码引起的。参照完整性规则要求，外码可以取参照关系中主码的已存在的值，也可以为空，外码值可以重复，但不能取参照关系中主码中不存在的值。例如，向“成绩表”添加记录时，添加的学号必须是“学生信息表”中已经存在的学号，也就是说，必须有这个学号的学生，才能输入这个学生的成绩，这符合实际的需求。

在数据库管理系统中，系统为用户提供了设置参照完整性约束的环境和手段，通过系统自身和用户自定义的完整性约束，可以充分保证关系的完整性、相容性和正确性。

**4. 关系运算**

用户可以使用关系运算从关系中查询所需要的数据，关系运算主要包括选择、投影和连接等。

(1) 选择。从关系中筛选出满足给定条件的元组的操作称为选择，选择是从行上进行选择。如在学生关系中查询所有女生的信息，就是一个关系选择运算。

(2) 投影。从关系中指定若干个属性组成新的关系称为投影，投影是从列上进行选

择。如在学生关系中显示所有学生的学号、姓名和出生日期，就是一个关系投影运算。

(3) 连接。连接是把两个关系中的记录按一定的条件进行连接，生成一个新的关系。不同关系中的公共属性是实现连接运算的纽带。如利用学生关系和成绩关系显示所有学生的学号、姓名、课程名和成绩，就是一个关系连接运算。

**5. 关系的规范化**

为了使数据库的设计方法趋于完备，人们研究了规范化理论。规范化理论认为，关系数据库中的每一个关系都要满足一定的规范。根据满足规范的条件不同，可以划分为5级，从第一范式(1NF(Normal Forms))～第五范式(5NF)，其中第一范式为最低级范式。

一个低级范式的关系通过各种方法可转换成多个高一级范式的关系，这个过程称为**规范化**。如果一个关系没有经过规范化，可能会出现数据冗余、数据更新不一致、数据插入和删除异常等现象。在实现一般的数据库应用时，通常只要把数据表规范到第三范式就可基本满足需求。

(1) 第一范式(1NF)。如果一个关系的所有属性都是不可再分的基本数据项，则该关系满足第一范式。第一范式是最低的规范化要求，是每个关系必须要满足的。

(2) 第二范式(2NF)。如果一个关系满足1NF，且所有的非主关键字都完全依赖于主关键字，则该关系满足第二范式。例如成绩(学号，课程号，课程名，学分，成绩)，属性“课程名”、“学分”只完全依赖于“课程号”，不完全依赖于“学号”，所以它不满足2NF。

可以用投影分解的方法使它满足2NF。如上面的成绩关系，可分解成两个关系：

① 成绩(学号，课程号，成绩)

② 课程(课程号，课程名，学分)

这样每个关系就满足2NF了。

(3) 第三范式(3NF)。如果一个关系满足2NF，且所有的非主关键字都不传递依赖于主关键字，则该关系满足第三范式。如学生(学号，姓名，性别，系编号，系名，系主任，系办电话)，“系编号”依赖于“学号”，而(系名，系主任，系办电话)等属性依赖于“系编号”，这称为传递依赖，不满足3NF。

同样也可以用投影分解的方法使它满足3NF。如上面的学生关系，可分解成两个关系：

① 学生(学号，姓名，性别，系编号)

② 系(系编号，系名，系主任，系办电话)

这样每个关系就满足3NF了。

一般地，关系模型的规范化都是通过投影分解来实现的，分解时应注意满足下列3个条件。

(1) 无损连接分解，分解后不丢失信息。

(2) 分解后每个关系都是高一级范式，不要同级分解或低级分解。

(3) 分解后的关系数量应做到尽量少。

对关系进行规范化，可以避免大量的数据冗余、节省空间和保持数据的一致性等。

## 1.3　关系数据库的设计

关系数据库是基于关系模型的数据库。在关系数据库中数据被分散到不同的数据表中保存，以便使每一个表中的数据只记录一次，减少数据冗余，其数据结构简单、清晰，易于操作和管理。它既解决了基于层次模型的数据库横向关联不足的缺点，又避免了基于网状模型的数据库关联过于复杂的问题，是目前发展最快、应用最广泛的数据库。

### 1.3.1　关系数据库

由关系模型建立的数据库称为关系数据库。关系数据库是由多个二维表组成的。

关系模型与关系数据库中的术语对应关系如表 1.3 所示。

**表 1.3　关系模型与关系数据库中的术语对照**

| 关系模型 | 关系数据库 | 关系模型 | 关系数据库 | 关系模型 | 关系数据库 |
|---|---|---|---|---|---|
| 关系 | 表 | 属性 | 字段 | 外码 | 外键 |
| 元组 | 记录 | 主码 | 主键 | | |

**1. 表**

在关系数据库中，表是基本的存储数据单位，每个表之间具有独立性(用表名来区分每一个表)，而且多个表之间具有相关性，这样使得数据的操作方式简单和方便。

在表中，数据的保存形式类似于电子表格，是以行和列的形式保存的，表中的行和列分别称为记录和字段。

若表的一个字段或几个字段的组合能够标识一条记录，则称其为关键字；当一个表中有多个关键字，但在实际应用中只能选择一个，被选用的关键字称为主键。

在关系数据库中，表之间是相互关联的。两个表通过主键和外键作为纽带建立关联关系，将主键所在的表视为**主表**，将外键所在的表视为**从表**。例如，在学生信息表和学生成绩表通过“学号”建立关联，其中学生信息表是主表，学生成绩表为从表。

从表中的外键值或为空，或者是主表中的主键值，所以我们称主表为外键的**参照表**。如学生信息表为学生成绩表中“学号”的参照表。

**2. 关系数据库的特点**

(1) 以面向系统的观点组织数据，使数据具有最小的冗余度，支持复杂的数据结构。

(2) 具有高度的数据和程序的独立性，应用程序与数据的逻辑结构及物理存储方式无关。

(3) 数据库中的数据具有共享性，能为多个用户提供服务。

(4) 关系数据库允许多个用户同时访问，同时提供了多种控制功能，能保证数据的安全性、完整性和并发性控制。

### 1.3.2 关系数据库的设计步骤

在数据库应用系统的开发过程中，数据库设计是开发的核心和基础。数据库设计是指对于一个给定的应用环境，构造最优的数据模式，建立数据库及其应用系统，使创建的数据库能够有效地存储数据，满足各类用户的应用需求。

设计数据库，一般需要遵循以下 6 个步骤。

**1. 需求分析**

需求分析阶段是数据库设计的第一步，也是后继各阶段的基础，是最为困难、最耗费时间的阶段。需求分析是否准确直接关系到数据库的成败和质量，影响到数据库应用系统的开发和使用。

需求分析阶段的主要任务是从多方面对整个组织进行调查，大量收集基础数据，全面了解用户对系统的信息需求、处理需求、安全性和完整性需求。需求分析人员要求既要懂数据库技术，又要对具体业务熟悉，一般由数据库专业人员与领域专家合作进行。

**2. 概念结构设计**

概念结构设计是把用户需求分析进行综合、归纳、抽象，建立一个独立于具体的 DBMS、独立于具体实现的概念模型。

概念模型通常使用 E-R 图描述数据及数据之间的联系，所以概念模型也称为 E-R 模型。E-R 图的具体描述请参见 1.2.1 小节。

**3. 逻辑结构设计**

逻辑结构设计就是将概念模型转换为与选用 DBMS 支持的数据模型相符的逻辑结构，并且对逻辑结构进行优化。

概念模型转换成关系模型，是将一个实体型转换成关系模型，转换过程中要做到不违背关系的完整性约束，尽量满足规范化原则。

概念模型向关系模型转换的原则如下。

(1) 每一个实体型转换为一个关系，实体的属性就是关系的属性，实体的关键字就是关系的主码。如图 1.7 中两个实体型学生和课程转换为两个关系：

学生(学号，姓名，性别，出生日期，政治面貌，班级)

课程(课程号，课程名，课程性质，学分)

(2) 联系的转换。

① 一般 1∶1、1∶$n$ 联系不产生新的关系，而是将一方实体的关键字加入到多方实体对应的关系中，联系的属性也一并加入。如图 1.6(b)中班级和学生的联系，直接将班级的关键字加入到学生关系中即可体现。

② $m$∶$n$ 联系产生新的关系，新关系由联系所涉及的实体的关键字加上联系的属性组成。如图 1.7 中的联系转换为成绩表(学号，课程号，成绩)。

(3) 关系模型的优化。应用关系的规范化理论对上述产生的关系进行规范化。关系规范化的具体描述请参见 1.2.3 小节。

**4. 物理结构设计**

在逻辑结构设计的基础上，选取一个最适合应用环境的物理结构和存储方法。

(1) 确定合适的数据库管理系统(DBMS)。物理结构是面向特定的 DBMS 的，必须首先确定使用的 DBMS，可以从 Access、SQL Server、MySQL、Oracle 等常用的关系 DBMS 中进行选择。

(2) 确定数据库的物理结构。在设计数据库的物理结构时，要了解选用的 DBMS 的功能、熟悉存储设备的性能。对关系数据库而言，要根据逻辑结构创建数据表、确定索引方式、确定完整性的约束条件、确定最佳性能的数据存储结构和数据访问方式，并为应用系统设置安全保护措施。

(3) 对物理结构进行评价。从时间效率、空间效率、维护开销和各种用户需求等方面进行权衡，从多个设计方案中选择一个较优的方案。

**5. 系统实施**

将物理结构设计的结果，创建一个具体的数据库，录入原始数据到数据库中，编写 DBMS 能够接受的应用程序，对数据库进行调试和测试。

要创建一个 Access 数据库，具体步骤如下。

(1) 创建一个空数据库。

(2) 创建数据库中的表。

(3) 确定表的主键。

(4) 建立各表之间的关联关系。

(5) 录入原始数据。

(6) 创建其他数据库对象。

**6. 系统运行与维护**

数据库系统正式投入运行后，为了保证运行良好，需不断地对数据库进行维护；而且只要数据库存在，维护工作就会一直进行下去。

对数据库经常性的维护工作主要由数据库管理员(DBA)完成。维护工作一般包括以下内容。

(1) 数据库的转储和恢复。

(2) 数据库的安全性和完整性控制。

(3) 数据库性能的监督、分析与改进。

(4) 数据库的重组织和重构造。

## 习　题　1

**一、选择题**

1. Access 数据库的类型是(　　)。

A. 层次数据库　　B. 网状数据库　　C. 关系数据库　　D. 面向对象数据库

2. E-R 图是数据库设计的工具之一，它适用于建立数据库的(　　)。

A. 概念模型　　B. 逻辑模型　　C. 结构模型　　D. 物理模型

3. 一个学生可以同时借阅多本书，一本书只能由一个学生借阅，学生和图书之间为(　　)联系。

A. 多对多　　B. 一对一　　C. 一对多　　D. 多对一

4. 数据库管理系统(DBMS)是(　　)。

A. 数据库应用系统　　B. 一组硬件

C. 软件集合　　D. 既有硬件也有软件

5. 在数据库中能够唯一地标识一个记录的字段或字段组合称为(　　)。

A. 记录　　B. 字段　　C. 域　　D. 主键

6. 关系数据库中的表不必具有的性质是(　　)。

A. 数据项不可再分　　B. 同一列数据项要具有相同的数据类型

C. 记录的顺序可以任意排列　　D. 字段的顺序不能任意排列

7. 关于主关键字(即主键)的说法正确的是(　　)。

A. 作为主关键字的字段，它的数据能够重复

B. 主关键字段中不许有重复值和空值

C. 一个表可以设置多个主关键字

D. 主关键字只能是单一的字段

8. 二维表由行和列组成，每一行表示关系的一个(　　)。

A. 属性　　B. 字　　C. 集合　　D. 元组

9. 下面关于关系描述错误的是(　　)。

A. 关系必须规范化

B. 在同一个关系中不能出现相同的属性名

C. 关系中允许有完全相同的元组

D. 在一个关系中列的次序无关紧要

10. 关系数据库系统能够实现的 3 种基本关系运算是(　　)。

A. 索引，排序，查询　　B. 建库，输入，输出

C. 显示，统计，复制　　D. 选择，投影，连接

11. 要从学生关系中查询 20 岁的女生所进行的查询操作属于(　　)。

A. 选择　　B. 投影　　C. 联结　　D. 自然联结

12. 要从学生关系中查询学生的姓名和班级，则需要进行的关系运算是(　　)。

A. 选择　　B. 投影　　C. 连接　　D. 求交

13. 下列叙述中正确的是(　　)。

A. 数据处理是将信息转化为数据的过程

B. 数据库设计是指设计数据库管理系统

C. 如果一个关系的属性或属性集并非该关系的主码，但它是另一个关系的主码，则称其为本关系的外码

D. 关系中的每列称为元组，一个元组就是一个字段

14. 在数据库中存储的是(　　)。

A. 数据　　B. 数据模型

C. 数据以及数据之间的关系　　D. 信息

15. 在分析建立数据库的目的时应该(　　)。

A. 将用户需求放在首位　　B. 确定数据库结构与组成

C. 确定数据库界面形式　　D. 以上所有选项

16. 数据库物理设计完成后,进入数据库实施阶段,下列各项中不属于实施阶段的工作是(　　)。

A. 创建数据库　　B. 扩充功能　　C. 输入数据　　D. 系统调试

## 二、填空题

1. 数据处理技术经历了人工管理、文件管理和__【1】__这3个发展阶段。

2. 数据库系统的主要特点为数据的结构化、__【2】__、易扩充、冗余度低、__【3】__和数据控制功能较强。

3. 两个实体之间的联系有3种,分别是一对一联系、__【4】__联系和__【5】__联系。

4. 关系中的行称为__【6】__,列称为__【7】__。

5. 在关系数据库中,唯一标识一条记录的一个或多个字段称为__【8】__。

6. 参照完整性是一个准则系统,Access使用这个系统用来确保相关表中记录之间__【9】__的有效性,并且不会因意外而删除或更改相关数据。

7. 在表中,数据的保存形式类似于电子表格,是以行和列的形式保存的。表中的行和列分别称为记录和字段,其中,记录是由一个或多个__【10】__组成的。

8. 在关系数据的基本操作中,把两个关系中相同属性值的元组连接到一起形成新的二维表的操作称为__【11】__。

9. __【12】__是指关系模型中的每一个关系模式都必须满足一定的要求。

## 三、思考题

1. 简述数据库系统的组成。

2. 什么是实体?什么是属性?在Access的数据表中,它们被称作什么?

3. 什么是主键?什么是外键?试举例说明。

4. 什么是数据库?数据库管理系统的功能是什么?

5. 设有一个学习关系:

*R*(学号、姓名、课程号、课程名、成绩、任课教师编号、教师名、职称)

假设每门课程只能有一个任课教师,试回答:

(1) 写出关系*R*存在的函数依赖关系。

(2) *R*属于第几范式?说明理由。

(3) 如果*R*不是3NF,将其分解为3NF,并设置分解后每个关系的主键,并写出它们之间联系的种类。