

第 5 章

Access 2010 数据库管理

随着计算机和互联网技术的不断发展，人们面临要存储、分析和处理的信息也越来越多，越来越复杂。在计算机中信息是通过数据来表示的，数据是信息的表现形式。数据可以是数字、字符、声音、图像、视频等多种形式。

数据库技术产生于 20 世纪 60 年代末 70 年代初，是计算机领域的重要分支之一。数据库技术主要研究如何利用计算机对数据进行科学有效的分类、组织、编码及存储，从而实现对数据的高效检索与维护，并提高数据的低冗余、高共享、高安全性。Access 2010 由微软公司推出的一个小型的数据库管理系统软件。本章将主要介绍数据库的基础知识、数据库的基本操作和 Access 2010 数据库管理系统软件的使用方法。

5.1 数据库基础知识

5.1.1 数据库管理技术的发展

数据 (Data) 是对客观事物特征和性质的符号表示，是对现实世界的抽象，这些符号包括文字、图形、声音、视频等。

数据处理是对各种类型的数据进行收集、存储、分类、计算、加工、检索和传送的过程。

数据管理技术经过 50 多年的发展大概可以划分为 3 个阶段：人工管理阶段、文件系统阶段和数据库系统阶段。

1. 人工管理阶段

人工管理阶段主要指 20 世纪 50 年代中期以前。这个时期计算机主要用于科学计算，计算机的硬件及软件条件相对比较落后。人工管理阶段数据主要是通过人工进行收集、整理及存储和处理，因此在数据管理方面存在很多缺陷，主要表现在数据不能保存、冗余大、不共享、无独立性、无结构。

2. 文件系统阶段

文件系统阶段主要指 20 世纪 50 年代末到 60 年代中期，计算机硬件和软件都有了一定的发展，计算机也不再局限于科学计算，而是越来越多应用到数据管理领域。这个阶段的数据管理主要由文件管理系统完成，相对人工管理阶段数据在管理方面提高了有效性且可以长期保持，但仍然存在数据共享性差、冗余度大、独立性差等缺陷。

3. 数据库系统阶段

20 世纪 60 年代末数据管理技术进入数据库系统阶段。该阶段出现了大量的专门数据管理软件，应用于数据管理工作，即数据库管理系统，其中 Access 2010 是由微软公司开发的一种小型的数据库管理软件。与人工管理阶段和文件系统阶段相比，数据库系统阶段存在数据冗余度小、共享性高、独立性强、具有结构化，安全性高等优点。

5.1.2 数据库相关概念

1. 数据

数据 (Data) 是信息的载体，是描述事物的属性及特征的符号，可以是文字、图形、声音、

视频等形式。

2. 数据库

数据库 (DataBase, DB) 顾名思义是指存储数据的仓库, 是长期存储在计算机中的, 有组织的、大量的、可共享的数据的集合。

3. 数据库管理系统

数据库管理系统 (DataBase Management System, DBMS) 是位于用户与操作系统之间的数据管理软件。数据库管理系统主要包括数据定义、数据操作、数据的组织、存储与管理以及数据库的建设与维护等功能。

4. 数据库系统

数据库系统 (DataBase System, DBS) 是指由计算机硬件系统、计算机软件系统、数据库管理系统以及相关的软件和数据库管理员、用户等组成的系统。

数据库系统的特点:

- (1) 数据冗余度低。
- (2) 数据共享性高。
- (3) 数据独立性高。
- (4) 使用特定的数据模型。
- (5) 具有统一的数据控制功能。

5. 数据库应用系统

数据库应用系统 (DataBase Application System, DBAS) 是利用数据库管理系统软件开发的, 应用于特定公司或部门的某种需求的应用程序软件。

◎ 温馨提示

通常在不引起歧义的情况下，数据库管理系统常称为数据库。

5.2 数据模型

模型是对现实世界特征的模拟与抽象，而数据模型是将复杂的现实世界抽象并转化为计算机能够识别的形式。数据库是利用数据模型实现对抽象并转化后的现实世界数据信息进行存储和处理。作为数据模型，应满足 3 方面的基本要求：

- (1) 对现实世界进行真实地模拟和抽象。
- (2) 容易为人所理解。
- (3) 便于在计算机中实现。

5.2.1 数据模型的组成

数据模型通常包括数据结构、数据操作和完整性约束 3 个要素。

(1) 数据结构：对计算机中数据的组织方式以及数据之间的关系进行描述，是数据库静态特征的描述。

(2) 数据操作：数据库中各类数据对象所允许执行的操作集合，包括操作方法和操作规则等，是数据库动态特征的描述。

(3) 完整性约束：数据库中数据状态及状态变化过程中遵行的一组完整性约束条件（规则）的集合，是数据库中数据正确性和一致性的保障。

5.2.2 数据模型的分类

数据模型按照数据抽象的层次不同和面向的对象不同可以分为概念数据模型、逻辑数据模型和物理数据模型 3 种。

(1) 概念数据模型。概念数据模型主要面向用户，是用户和数据库设计人员之间沟通的手段。概念数据模型的特点是：用户容易理解，对现实世界特征的数据抽象，与具体的 DBMS 无关。常用的概念数据模型是实体—联系模型，简称为 E-R 模型。

(2) 逻辑数据模型。数据库设计人员利用 DBMS 设计数据库系统时所搭建的数据模型。常用的逻辑数据模型包括网状数据模型、层次数据模型、关系数据模型和面向对象数据模型等。目前，使用最广泛逻辑数据模型是关系模型。逻辑数据模型特点是：与用户无关，DBMS 可以识别与支持。

(3) 物理数据模型。物理数据模型描述的是数据在计算机内组织结构的数据模型，是最底层的数据模型。特点是：与 DBMS 有关，与操作系统和硬件密切相关。

5.2.3 实体-联系 (E-R) 模型

E-R 模型是常用的概念数据模型，用于数据库设计人员与用户之间沟通的主要模型，便于数据库设计人员了解用户的具体需要。E-R 模型中涉及的基本概念主要有实体、属性、实体型、实体集、关键字、域、联系等。

(1) 实体。客观存在且可以相互区别于彼此的个体为实体，如一名学生、一门课程、一座城市等都是实体。

(2) 属性。每个实体都具备若干特征，通过特征对其进行描述，我们将实体的特征称为实体的属性。例如，每个学生实体都具有学号、姓名、性别、出生年月等属性。

(3) 实体集。具有相同属性的一类实体的集合称为实体集。例如，全体学生、所有课程都是实体集。

(4) 实体型。用于描述实体集中所有实体所共同具有的属性特征。例如，学生（学号，姓名，性别，出生年月）就是一个实体型。

(5) 关键字。能够唯一标识一个实体的属性或属性集称为实体型的关键字。例如，每个学生实体的学号属性值互不相同，可以用于唯一标识一个学生实体，因此学号是学生实体型的关键字。

(6) 域。实体属性的取值范围被称作域，是一个集合。例如，性别的域为{男，女}。

(7) 联系。客观世界中任何个体都不是孤立的，彼此间存在着一定的关系。在 E-R 模型中将实体之间的关系称为联系，以反映客观事物之间的关系，两个实体集之间的联系包括一对一 (1:1)，一对多 (1:n) 和多对多 ($m:n$) 3 种。

一对一联系 (1:1)：如果对于实体集 A 中的每个实体，实体集 B 中至多有一个（也可以没有）实体与之对应，反之亦然，则称实体集 A 与实体集 B 是一对一联系。如班级和班长之间的联系是一对一联系，如图 5-1 (a) 所示。

一对多联系 (1:n)：如果对于实体集 A 中的每个实体，实体集 B 中有 n 个实体 ($n \geq 0$) 与之对应；反之，对于实体集 B 中的每个实体，实体集 A 中至多有一个实体与之对应，则称实体集 A 与实体集 B 是一对多联系。如班级和学生之间属于一对多联系，如图 5-1 (b) 所示。

多对多联系 ($m:n$)：如果对于实体集 A 中的每个实体，实体集 B 中有 n 个实体 ($n \geq 0$) 与之对应；反之依然，则称实体集 A 与实体集 B 是多对多联系。如学生和课程之间属于多对多联

系, 如图 5-1 (c) 所示。

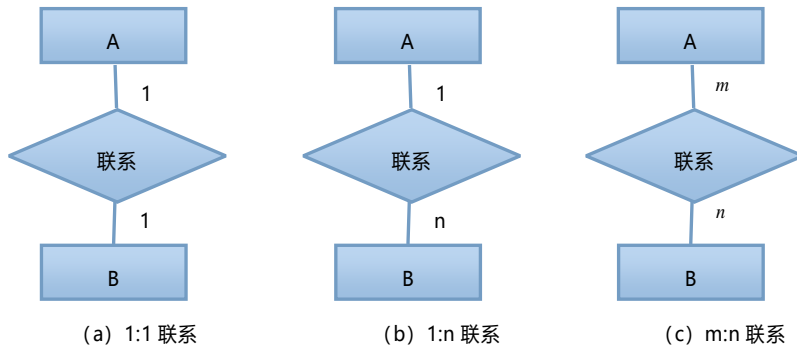


图 5-1 两个实体集之间的联系

◎ 温馨提示

1:1 联系属于特殊的 1:n 联系, 1:n 联系属于特殊的 m:n 联系。

5.2.4 关系数据模型

当前数据库管理系统中应用广泛的逻辑数据模型是关系模型。关系模型即二维表, 用一张二维表描述实体集以及实体集之间的联系。采用关系模型搭建的数据库称为关系数据库。下面介绍关系数据模型中几个重要的概念。

关系: 在关系模型中, 一个关系就是一张二维表, 每个关系都有一个关系名。

元组: 将二维表中的行称为元组, 每一行是一个元组, 代表一个实体。

属性: 将二维表中的列称为属性, 每一列有一个属性名。

域: 属性的取值范围称为该属性的域。

关键字: 在二维表中, 值可以唯一标识一个元组的属性或属性组称为关键字。

外关键字: 如果一个属性或属性组不是关系 A 的关键字, 但是关系 B 的关键字, 则称该属性或属性组为关系 A 的外关键字或外码。

候选码: 一个关系中可能包括多个属性或属性组的值均可以唯一标识一个元组, 则这些属

性或属性组称候选码，选择一个作为主键，主键的属性称为主属性。

5.2.5 关系运算

在关系数据库中，关系的基本运算有两类：传统的集合运算和专门的关系运算。

1. 传统的集合运算

设有两个关系 R 和 S ，它们具有相同的结构（即包括相同的属性），则关系 R 和 S 可做如下

3 种运算：

(1) 并

关系 R 和 S 的并是由属于关系 R 或关系 S 的元组构成的集合，并运算符用 \cup 表示。

(2) 交

关系 R 和 S 的交是由关系 R 和关系 S 共同具有的元组组成的集合，交运算符用 \cap 表示。

(3) 差

关系 R 和 S 的差是由属于关系 R 但不属于关系 S 的元组组成的集合。差运算符用 $-$ 表示。

关系 R 和 S 的并、交和差操作结果仍为关系，且结构与关系 R 和 S 相同，如图 5-2 所示。

A	B	C
a_1	b_1	c_1
a_1	b_2	c_2
a_2	b_2	c_1

关系 R

A	B	C
a_1	b_2	c_2
a_1	b_3	c_2
a_2	b_2	c_1

关系 S

A	B	C
a_1	b_1	c_1
a_1	b_2	c_2
a_2	b_2	c_1
a_1	b_3	c_2

$R \cup S$ 结果

A	B	C
a_1	b_1	c_1

$R - S$ 结果

A	B	C
a_1	b_2	c_2
a_2	b_2	c_1

$R \cap S$ 结果

图 5-2 传统的集合运算

2. 专门的集合运算

(1) 选择

选择运算是从给定关系中选出满足指定条件的元组构成的关系。从关系 R 中选出满足条件 F 的元组构成一个新关系，即对关系 R 做选择操作，选择条件为 F ，记作 $\delta_F(R)$ 。

(2) 投影

投影运算是从给定关系中选出某些属性列构成的关系。从关系 R 中选择 A 、 B 、 C 属性列构成一个新的关系，即对关系 R 做投影操作，投影到 A 、 B 、 C 属性列，记作 $\Pi_{A,B,C}(R)$ 。

(3) 联接

联接运算是从两个关系中选出同名属性列间满足指定关系的两个元组合并，合并后的结果作为结果关系中的一个元组。若关系 R 和 S 都包括属性列 A ，则关系 R 和 S 做连接条件为 $R.A=S.A$ 的连接操作，记作 $R \underset{R.A=S.A}{\bowtie} S$ 。

图 5-3 所示是专门的集合运算。

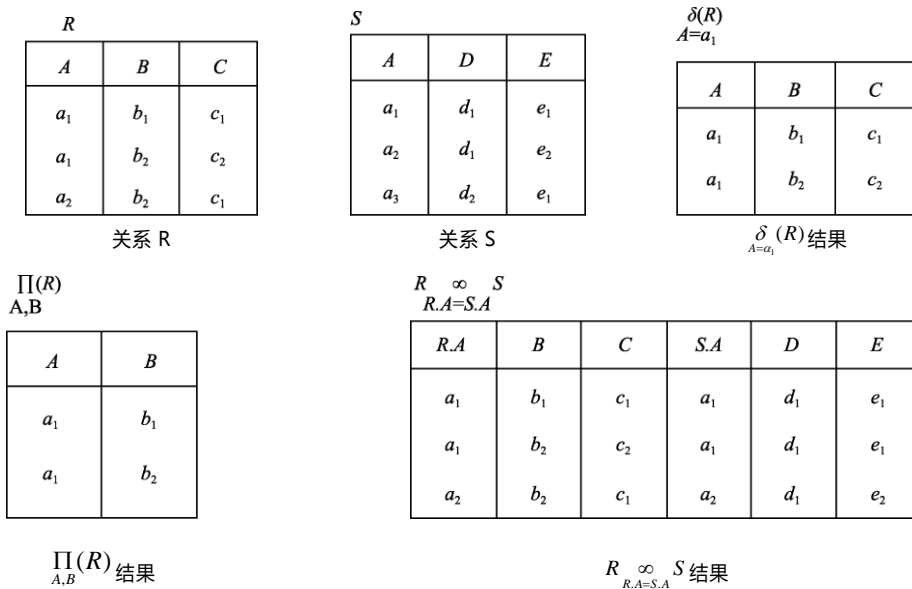


图 5-3 专门的集合运算

◎ 温馨提示

投影操作时，投影的属性列名可以用对应的属性列号替代。如关系 R ，则 $\Pi_{A,B}(R)$ 也可以记作 $\Pi_{1,2}(R)$ 。

5.2.6 数据库设计的基本步骤

数据库设计是一项涉及多样学科的综合性的技术，是一项庞大的工程项目。数据库设计的基本步骤一般可以分为如下 6 步：

本步骤一般可以分为如下 6 步：

1. 需求分析

收集与分析用户的需求，将用户的需求整理为数据字典，再用数据流图描述处理需求。

2. 概念结构设计

对需求进行综合归纳与抽象，形成一个独立于具体 DBMS 的概念模型（通常用 E-R 图表示）。

3. 逻辑结构设计

将概念模型转换为某一个 DBMS 所支持的数据模型（如关系模型），并对其进行优化。

4. 数据库物理设计

为逻辑数据模型选取一个最佳应用环境的物理结构（包括存储结构和存取方法）。

5. 数据库实施

运用 DBMS 提供的语言（如 SQL）及其宿主语言（如 C 语言），根据逻辑结构和物理结构建立具体的数据库，编码与调试应用程序，并组织数据入库。

6. 数据库运行与维护

运行数据库应用系统。在数据库应用系统运行过程中，还需要不断的检测、修改与维护。