

第 1 章 数据库简介

随着人类社会的发展和科技的进步,数据挖掘、大数据分析、云计算、人工智能、区块链等新兴技术成了时下的热门技术,这些新兴的热门技术无一不将数据作为核心内容,因此,海量数据的有效组织、存储、处理和共享变得尤为重要,这也促使了数据库技术的不断更新和快速发展。学好数据库相关技术和基础知识,可以为今后更深入地学习或就业奠定重要的基础。

本章主要内容如下:

- 了解课程的定位及掌握数据库的基础知识。
- 了解数据库的发展阶段、SQL 语言命令和数据库访问接口。
- 熟悉数据模型及其分类。
- 掌握利用 E-R 图描述概念模型。
- 重点掌握将 E-R 图转换成关系模型。
- 熟悉关系数据库的设计步骤。

【相关单词】

- | | |
|----------------------|---------------------|
| (1) data: 数据 | (2) information: 信息 |
| (3) database: 数据库 | (4) entity: 实体 |
| (5) relationship: 关系 | (6) attribute: 属性 |
| (7) key: 键 | (8) model: 模型 |

1.1 了解数据库的基本知识

熟练掌握数据库的基础知识,了解数据库技术的发展阶段、SQL 语言命令和数据库访问接口等,对数据库技术和后续知识及内容的学习极为重要。

1.1.1 课程定位

1. 为什么要学习“MySQL 数据库原理与应用”课程

“MySQL 数据库原理与应用”课程是计算机应用技术相关专业的一门专业必修课,是程序开发、网络程序设计的基础课程之一。通过对“MySQL 数据库原理与应用”课程

的学习,能够提高读者操作和使用数据库的能力;结合 PHP、Java 等程序设计语言的学习,能够进行软件的开发和网站的建设,承担软件编码或者软件测试等相关工作。

2. “MySQL 数据库原理与应用”课程主要学习的知识

课程包括以下内容:数据库的基础知识,MySQL 数据库的安装与配置,MySQL 数据库的数据类型,数据库的创建、删除及存储引擎,数据库的查询,数据库中运算符的使用,数据库函数的运用,数据库的存储过程、触发器及索引,数据库视图及数据库权限的管理。

3. 怎么学习“MySQL 数据库原理与应用”课程

MySQL 数据库中,数据类型、运算符、数据库的基本操作及权限管理是需要掌握的基础知识要点,在此基础上,数据库的查询、函数的使用、存储过程、触发器、索引、视图也是必要的知识。编者认为,不能仅仅学习数据库知识而忽略与它相关的程序设计语言和相关主流软件开发平台,因此建议读者在学习“MySQL 数据库原理与应用”课程时结合至少 1 门程序设计语言进行学习。

1.1.2 数据库的相关知识点及概念

数据库由数据库管理员操作数据库管理系统并完成数据的存储,以便后期其他程序可以对数据进行传输、分析和应用。

1. 信息

现阶段正处于一个信息爆炸的时代,所有的人类活动都离不开信息。1948 年,数学家香农曾提出:“信息是用来消除随机不定性的东西。”

控制论创始人维纳提出了信息的经典性定义:“信息是人们在适应外部世界,并使这种适应反作用于外部世界的过程中,同外部世界进行互相交换的内容和名称。”

不同的专业领域对信息的定义不一样,科学的信息定义是:信息是对客观世界中各种事物的运动状态和变化的反映,是客观事物之间相互联系和相互作用的表征,表现的是客观事物运动状态和变化的实质内容。

提前查看天气变化的信息,可以决定是否带伞;查看公交车到站时间的信息,可以决定何时出门;查看商家的促销信息,可以决定何时购买商品更划算等。这都是人类利用信息解决日常生活问题的依据。

2. 数据

数据(data)是对客观事件进行记录并便于人们鉴别的符号,是对客观事物的性质、状态以及相互关系等进行记载的物理符号或这些物理符号的组合,它是可识别的、抽象的符号。数据不仅指数字,还包含某些具有特定意义的文字、字母、数字符号的组合、图形、图像、视频、音频等,也是客观事物的属性、数量、位置及其相互关系的抽象表示。

信息与数据的区别和联系：信息是数据的内涵，是对数据做有意义的解释；数据则生动具体地表达出信息。信息是对数据进行加工处理之后所得到的并对决策产生影响的数据，具有逻辑性和观念性。数据本身没有意义，数据只有对实体行为产生影响时才能成为信息。例如商家依据客户购买习惯、经济能力、年龄等数据，建立不同客户的人群肖像，然后根据不同的消费群体情况，对不同的商品有针对性地发布促销活动和个性化推荐信息。

连续的数值可以是声音或者图像等，称作模拟数据；离散的数值，如符号、文字等，称作是数字数据。

在计算机中，用二进制数制中的 0 和 1 组成的数字序列表示数据。

3. 数据库

数据库(Database, DB)是存放数据的仓库，它是以一定结构方式存储在计算机内的、有组织的、能供不同用户共享的、能够长期存储且具有尽可能小的冗余度、与应用程序之间彼此相互独立的数据集合。数据库具有集成性、海量性、共享性和持久性等特点，用户能够对数据库中的数据进行新增、查询、更新、删除等操作。

4. 数据库管理员

数据库管理员(Database Administrator, DBA)是对从事管理和维护数据库管理系统(DBMS)的相关工作人员的统称。DBA 主要负责业务数据库从设计、测试到部署交付的全生命周期管理，管理的核心目标是保证数据库管理系统的稳定性、安全性、完整性和高性能。数据库管理员也称作数据库工程师(Database Engineer)，主要职责是保证数据库在 7 天×24 小时内稳定高效地运转。数据库工程师与数据库开发工程师(Database Developer)不同，前者主要负责数据库管理系统的运维管理，后者则从事设计和开发数据库管理系统和数据库应用软件系统的工作，侧重于软件开发。

5. 数据库管理系统

数据库管理系统(Database Management System, DBMS)是一种操纵和管理数据库的大型软件，用于建立、运用、管理和维护数据库。它对数据库进行统一的管理和控制，以保证数据库的安全性和完整性，能够提供数据录入、修改、查询，具有数据定义、数据操作、数据存储与管理、数据维护、通信等功能，且能够允许多用户使用。主要特性是：控制并减少数据冗余度；保证数据的独立性和一致性；提高系统的安全保密性，并实现数据的共享。

1.1.3 数据库技术的发展阶段

计算机及网络技术的不断进步，使得以数据为核心的新兴技术正经历着不断的更新和变革，这也让数据库技术得以快速发展和广泛应用。通常将数据库技术划分为人工管理、文件管理、数据库管理和高级数据库管理四个阶段。

1. 人工管理阶段

计算机诞生之初,在数据管理方面以手工方式为主,用纸卡及报表等进行数据的记录、存放、查询和更新。当时没有磁盘或其他直接存取的外存设备,更没有操作系统和数据库,因此工作效率低下。人工管理数据阶段的特点如下:计算机不存储数据;数据面向应用;数据不独立;无数据文件处理软件。

2. 文件管理阶段

随着计算机的发展,电子管逐渐被晶体管所取代,存储介质也发生了改变,数据得以以普通文件或二进制文件的形式保存,操作系统、汇编语言和一些高级语言也逐渐出现。计算机不仅用于科学计算,还大量应用于管理,其中文件管理系统是操作系统中专门的数据管理软件,这是数据库系统的雏形,但并不是真正的数据库系统。文件系统管理数据的特点如下:数据可以长期保存;具有简单的数据管理功能,但数据的独立性、共享性差,冗余大,数据不一致,数据联系弱。

3. 数据库管理阶段

伴随着计算机软硬件技术的革新,大规模集成电路在 CPU 上得以实现,使得数据库在存储和处理海量数据方面的技术得到了极大的提升。各种数据库管理系统的大量涌现让数据库管理技术得到了不断的更新和完善,对计算机领域产生了巨大的影响并形成了“数据库时代”。数据库管理阶段的特点如下:数据的集成性、独立性高,数据的冗余度低,共享性强,可以对数据进行统一的管理和控制。

4. 高级数据库管理阶段

数据库技术在商业领域中取得的巨大成功,让其他领域对数据库技术的需求与日俱增。以数据为核心的新兴应用领域极大地推动了数据库技术的发展,并使其不断与其他高新科技相结合,向更高级别的数据库技术发展和更新迭代,继而出现了分布式数据库技术、面向对象数据库技术、面向应用领域数据库技术等。根据数据库的应用及相关分析机构评估,数据库技术将面向应用、业务服务等方面,并不断为新型应用提供更多的技术支持。

1.1.4 数据库技术的构成

数据库系统主要由硬件部分和软件部分组成。硬件部分主要是存放数据库中的数据,包括计算机及相应的存储设备;软件部分则包含数据库管理系统及其运行的操作系统,以及支持多种语言进行软件程序开发的访问技术等。

1. 数据库系统

数据库系统(Database System,DBS)主要由数据库、数据库管理系统(DBMS)、数据

库应用程序(Database Application)三部分组成。数据库是存放数据的容器,一个数据库系统可以包含多个数据库,一个数据库可以包含多个数据文件。DBMS 介于用户和操作系统之间,负责数据库的创建、维护,并对数据库进行统一管理,保证数据库的安全性、完整性和可靠性。在大多数情况下,DBMS 无法满足对数据管理的要求,这时就需要借助数据库应用程序对数据库进行操作。数据库应用程序能够直接与 DBMS 进行通信、访问和管理 DBMS 中的数据,允许用户插入、修改、删除数据库中的数据,如图 1-1 所示。

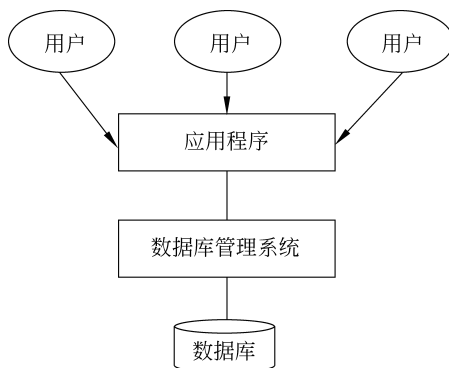


图 1-1 数据库系统

2. SQL

SQL 全称是 Structured Query Language(结构化查询语言),是对数据库进行查询和修改的语言。不同数据库厂商提供不同版本的 SQL,但都包含原始的 SQL 标准。SQL 主要包含以下四个部分。

(1) 数据定义语言(Data Definition Language,DDL): 主要作用是在数据库中创建(create)、修改(alter)、删除(drop)表。

(2) 数据操作语言(Data Manipulation Language,DML): 主要作用是对表中的数据进行插入(insert)、更新(update)和删除(delete)。

(3) 数据查询语言(Data Query Language,DQL): 主要作用是对数据库进行查询(select),并对查询结构按照指定要求操作(where、order by、group by 和 having)。

(4) 数据控制语言(Data Control Language,DCL): 主要作用是授权(grant)单个用户或用户组对数据库对象进行访问或回收(revoke)权限。

上面简单列举了操作数据库及表数据的相关命令,目的是让读者提前了解 SQL 查询语言,接下来将会在 MySQL 的学习过程中详细介绍这些知识及其使用方法。

3. 数据库访问接口

(1) ODBC

ODBC(Open Database Connectivity,开放数据库连接)是微软公司提出的基于 SQL 的一种用于访问数据库的应用程序通用编程接口标准。它介于 SQL 和应用程序界面之间,解决了数据库发生改变时应用程序随之改变的问题。利用 ODBC 接口生成的应用程序能够保证与数据库或数据库引擎无关,从而达到为使用数据库的用户和从事软件开发的人员屏蔽复杂异构环境的目的。

ODBC 实际上是一个数据库访问函数库。基于 ODBC 的应用程序对数据库的操作不依赖于 DBMS,而是靠对应 DBMS 的 ODBC 驱动程序完成对数据库的所有操作。由于它提供了数据库统一的访问接口,无论是 MySQL、Access 还是 Oracle 数据库,均可通过 ODBC API(Application Programming Interface)实现数据库的访问,并实现了应用程序

的平台无关性和可移植性。

(2) JDBC

JDBC(Java Database Connectivity,Java 数据库连接)是 Java 应用程序与数据库之间连接的桥梁,是一组用 Java 语言编写的类和接口,通过 JDBC API 可以实现对多种主流关系数据库进行操作。

(3) ADO.NET

ADO.NET(ActiveX Data Objects)是微软公司开发设计的一组用于与不同数据源进行交互的面向对象的类库。通常情况下,数据源可能是数据库,也可能是 Excel 表格、XML 文件或者文本。不同的数据源采用的协议不同,对于不同的数据源必须采用相应的协议。使用 ODBC 协议的老旧数据源和使用 OLE DB 的新数据源以及不断出现的其他新数据源,都可以通过 ADO.NET 类库进行连接。

(4) PDO

PDO(PHP Data Object)是为 PHP 访问不同数据库而定义的一个轻量级、一致性的接口,它提供了一个数据抽象层,使得无论使用何种数据库,都可以通过一致的函数对数据库进行操作,这也是 PHP 5 中新增加的一项重要功能。

1.1.5 常见数据库简介

1. Oracle 数据库

Oracle Database 又称 Oracle RDBMS,简称 Oracle,它是美国 Oracle(甲骨文公司)的一款关系数据库管理系统。Oracle 数据库应用于大、中、小型机上,已成为世界上使用最广泛的关系数据库系统之一。其主要的特点是可移植性强,兼容性高,数据安全性高,支持各种协议,能够提供多种开发工具,极大地方便了用户进一步的开发。

2. MS SQL Server 数据库

MS SQL Server 数据库是由美国微软公司开发的一款关系数据库管理系统,常用于在 Web 上存储数据,广泛应用于银行、保险、电子商务等与数据库相关的行业。该数据库因友好的操作界面和易操作性等而深受用户喜爱。其特点有:具有直观的图形化界面,操作方便;拥有丰富的编程接口工具,方便用户进行程序设计;有很好的伸缩性,便于跨界运行,从微型计算机到大型处理器均可运行;能够支持 Web 技术,使得用户能够将数据库中的数据发布到 Web 上;符合真正的客户(Client)/服务器(Server)体系结构。

3. MySQL 数据库

MySQL 数据库是由瑞典 MySQL AB 开发的一款关系型数据库管理系统。其特点是:运行速度快、体积小;总体拥有成本低;特别是 MySQL 对用户开放源码这一特点,使得多数中小型网站的开发都选择它作为网站建设的数据库,搭配 PHP 和 Apache 可组成良好的建站开发环境。

4. DB2 数据库

DB2 数据库是由 IBM 公司开发的一款关系数据库管理系统。该数据库主要运行在大型应用系统上,具有较好的伸缩性,提供了高层次的数据完整性、安全性、恢复性和利用性,具有平台无关性的基本功能和 SQL 命令。应用程序能够通过微软公司开发的开放数据库连接(ODBC)接口、Java 数据库连接(JDBC)接口等对 DB2 数据库进行访问和操作。

1.2 数据库关系模型的设计

1.2.1 数据模型概述

1. 数据模型的基础知识

在客观现实世界中的任何事物及活动都具有一定的特征信息,将事物的特征和联系通过符号记录下来,并用规范化的语言描述现实世界中的事物,就构成了基于现实世界的信息世界。将具体事物及其之间的联系转换成计算机能够识别并处理的数据,需要用相关数据模型对信息进行建模(抽象、描述和表示)。数据由现实世界进入数据库中存储,通常需要经历现实世界、信息世界和计算机世界三个阶段,其主要转换过程如图 1-2 所示。

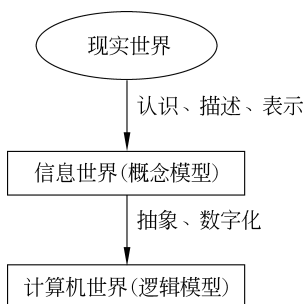


图 1-2 模型的抽象

(1) 现实世界。现实世界是客观存在的事物及其之间的联系。

(2) 信息世界。信息世界是对客观世界存在的事物及其之间的联系进行描述、抽象和表示,根据用户的角度对数据按一定规则进行建模(概念模型—实体与联系)。

(3) 计算机世界。计算机世界是建立在计算机上的模型,通过将信息世界中的具体事物及活动信息转换成计算机能够识别的数据,并按一定数据结构对数据进行存储。

数据是对客观事物的符号表示,模型是对现实世界的抽象,数据模型是客观世界数据特征的抽象。数据模型的三要素是数据结构、数据操作和完整性约束(数据的约束条件)。

2. 数据模型的分类

数据模型是对现实世界的抽象表示,通常需要符合三个条件:较好地模拟客观现实世界、容易理解和分析、在计算机中便于实现。

根据数据模型的具体应用,可将模型分为两类。

(1) 概念模型。概念模型又称为信息模型,介于现实世界与计算机世界之间,简单、清晰、易于用户理解的概念是建立概念模型的重要基础。概念模型是对现实世界的第一

层抽象,常作为数据库设计人员与用户进行交流和沟通的工具,是一种独立于计算机系统的数据模型。常用“实体联系图(E-R图)”来完成概念模型的设计。

(2) 逻辑模型。它包括关系模型、网状模型、层次模型、面向对象模型,是从计算机系统的角度对数据进行建模,根据数据库的逻辑结构对现实世界的第二层抽象。这类模型的设计直接与 DBMS 相关,称作“逻辑数据模型”,简称“逻辑模型”,又称“结构模型”。

1.2.2 概念模型

1. 概念模型的概念

从客观现实世界到信息世界的抽象过程中,有时只需要考虑数据本身的结构特征和相互之间的联系,暂不考虑计算机中数据的具体实现方式,通常采用概念模型进行分析。

(1) 实体

实体(entity): 在现实世界中可以相互区分的类别或事物,如一位学生、一个文件等。

实体集(entity set): 相同类别实体的集合,如一个班级的所有学生、一个商场的所有商品、一个科室的所有医生等都是相对应的实体集。

实体型(entity type): 相同类别实体共有特征的抽象表示。如学生类别的共有特征为学号、姓名、性别、年龄、专业、班级等。这些特征共同定义了学生这一类别的实体型,每个学生都具有这些特征,但每个学生具体的特征值既可以不同也可以相同,如性别、姓名等。对于同一类实体,抽象后的实体型特征可以不同,从而定义出不同的实体型。

实体值(entity value): 对实体型进行实例化(赋值),符合实体型的定义要求,是对实体的具体描述。

【例 1-1】 某大学教师的实体型可用姓名、性别、年龄、籍贯、民族、学历、职称等特征来描述,则李明、男、35、四川德阳、汉族、硕士、讲师就是一个实例化后的实体值,表示具体的一位教师个人的基本信息。在表 1-1 中,首行规定了教师的实体型,其后各行则表示实例化后的具体值。

表 1-1 教师基本信息表

姓 名	性别	年龄	籍 贯	民族	学历	职称
李明	男	35	四川德阳	汉族	硕士	讲师
赵亮	男	42	贵州贵阳	汉族	博士	教授
娜罕姆香	女	30	云南昆明	傣族	本科	助教
张伟	男	37	山东菏泽	汉族	硕士	副教授
丹妮雅	女	40	宁夏银川	回族	博士	教授
王杰	男	27	河南商丘	汉族	本科	助教
.....						

(2) 联系

联系(relationship): 实体之间存在的相互关系,通常表示为一种活动。例如,学生的一次选课、一场比赛、一张购物清单等都是联系。

联系集(relationship set): 相同类别联系的集合。如商场购物的所有清单、一次比赛活动中的全部比赛场次,一个班级同学的所有选课等。

联系型(relationship type): 对相同类别联系共有特征的抽象表示。

【例 1-2】 在学生选课关系中,选课编号、学号、课程编号、上课时间、上课地点、考试成绩等特征构成了联系型。其中,学号对应学生实体中的具体某位学生,课程编号对应课程实体中的具体某门课程。在表 1-2 中,首行对应联系型的所有特征,其后各行是对联系的实例化。

表 1-2 学生选课表

选课序号	学 号	课程编号	上课时间	上课地点	考试成绩
1	19080702	GG0047	周三 1~2 节	翰院 316	
2	19030522	GG1209	周一 5~6 节	贡院 424	
3	19060310	GG1546	周五 3~4 节	贡院 217	
4	19040806	GG5176	周二 7~8 节	艺步楼 516	
5	19060232	GG1096	周四 1~2 节	闻院 221	
6	19050109	GG2873	周一 9~10 节	翰院 215	
...					

(3) 属性、主键和域

属性(attribute): 描述实体或者联系中的某一特征。一个实体或联系通常具有多个特征,需要通过多个对应的属性进行描述和表示。实体的具体属性可根据实际需求决定,而非固定不变。

主键(primary key): 又称关键字、码或者关键码,是区别实体集中不同实体的唯一标识,如身份证号、员工编号、学生学号、电话号码等。一个实体可以有多个键。键在实体中可能是一个或多个属性,而在联系中通常为多个属性。例如,学生实体中以学号作为键,课程实体中以课程编号作为键,相对应的都是实体中的单一属性;在联系选课表中,以“学号”“课程编号”作为键才能标识一个联系值。作为键的属性称为主属性(main attribute),否则称为非主属性(non-main attribute)。

域(domain): 实体对应属性的取值范围,如“男、女”即是性别的域。

2. 联系分类

联系分类(relationship classify): 表示两个实体型之间联系的类型。根据一个实体型中的实体与另一个实体型中的实体所对应的关系,可分为一对一关系、一对多关系、多对多关系三种类型。

(1) 一对一关系

如果一个实体型中的一个实体至多与另外一个实体型中的实体产生联系,同样另一

个实体型中的一个实体至多与该实体型中的一个实体产生联系,则这两个实体型之间是一对一的关系,记为 $1:1$ 。例如,每个班只有一位班长,每位班长只服务一个班级。

【例 1-3】 在 $1:1$ 的联系中,两个实体可以属于同一实体型,也可以属于不同实体型。在公民实体型中的一位公民只拥有一个身份证信息,相应的一个身份证信息唯一对应一位公民,如图 1-3 所示。



图 1-3 一对一关系举例

(2) 一对多关系

如果一个实体型中的一个实体与另外一个实体型中的任意多个实体(包含 0 个)产生联系,同样另一个实体型中的一个实体至多与该实体型中的一个实体产生联系,则这两个实体型之间是一对多的关系,记为 $1:n$,与之相对应的是多对一关系。

【例 1-4】 在一个班级中,班级实体与学生实体之间的关系是 $1:n$ 关系。一个班包含多位学生,一个学生只属于一个班,如图 1-4 所示。

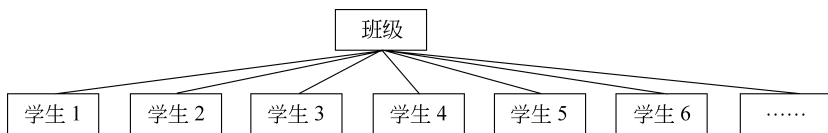


图 1-4 一对多关系举例

(3) 多对多关系

如果一个实体型中的一个实体与另外一个实体型中的任意多个实体(包含 0 个)产生联系,同样另一个实体型中的一个实体与该实体型中的任意多个实体(包含 0 个)产生联系,则这两个实体型之间是多对多的关系,记为 $m:n$ 。

【例 1-5】 在课程授课中,教师实体与学生实体的关系是 $m:n$ 关系,一位教师可以教授多位学生,一位学生可以上多位老师的课程,如图 1-5 所示。

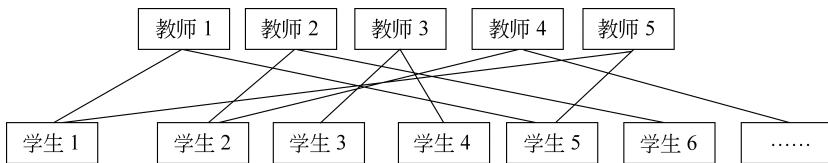


图 1-5 多对多关系举例

1.2.3 E-R 图的设计

1. E-R 模型的基本概念

实体联系模型(entity relationship model)也称 E-R 模型或实体—联系方法,用于描述实体及相互之间关系的概念模型。其简单易懂的表示方式和贴近生活化的描述特点,使之成为数据库设计人员与普通用户沟通交流和进行数据建模的常用工具。根据 E-R 模型生成的关系图称为 E-R 图。