



高职高专“十一五”规划教材

SQL Server 2008

数据库管理项目教程

张宝华 主编

兰 静 沈志梅 副主编



化学工业出版社

高职高专“十一五”规划教材

SQL Server 2008 数据库 管理项目教程

张宝华 主编

兰 静 沈志梅 副主编



化学工业出版社

· 北 京 ·

本书针对高职高专教学特点,从方便教和学的角度组织内容、精选实例并合理安排先后顺序,理论知识与项目化实例相结合,以 SQL Server 2008 的基本知识为主线,主要介绍了大型数据库管理系统 SQL Server 2008 的功能特点以及使用 SQL Server 2008 进行数据库开发的实用技术。

本书根据编者多年的教学和项目开发经验,每一部分均先讲解理论知识,然后是项目举例,最后是项目实训。各个项目实例既是单独的,又是可以将全书的项目贯穿组成大项目,比较好地解决了 SQL Server 2008 学 and 用的问题。全书比较系统地介绍了数据库基本知识、SQL Server 2008 数据库创建、数据库管理、数据查询、视图、索引、T-SQL、事务与游标、存储过程和触发器、数据库的备份和恢复、系统安全管理等内容,并给出了 ASP.NET 与 SQL Server 2008 联合开发编程实例。

本书既可作为大中专学生学习数据库课程的教材,又可以作为广大数据库应用开发人员的参考资料和培训教材。

图书在版编目(CIP)数据

SQL Server 2008 数据库管理项目教程 / 张宝华主编.

—北京: 化学工业出版社, 2010.7

高职高专“十一五”规划教材

ISBN 978-7-122-08651-8

I. S… II. 张… III. 关系数据库-数据库管理系统, SQL Server 2008-高等学校: 技术学院-教材
IV. TP311.138

中国版本图书馆 CIP 数据核字 (2010) 第 119701 号

责任编辑: 王听讲

装帧设计: 刘丽华

责任校对: 蒋 宇

出版发行: 化学工业出版社 (北京市东城区青年湖南街 13 号 邮政编码 100011)

印 装: 大厂聚鑫印刷有限责任公司

787mm×1092mm 1/16 印张 17¼ 字数 472 千字 2010 年 8 月北京第 1 版第 1 次印刷

购书咨询: 010-64518888 (传真: 010-64519686)

售后服务: 010-64518899

网 址: <http://www.cip.com.cn>

凡购买本书, 如有缺损质量问题, 本社销售中心负责调换。

定 价: 33.00 元

版权所有 违者必究

前 言

熟练掌握并灵活应用 SQL 是数据库初学者和数据应用开发人员必备的基本功。Microsoft SQL Server 2008 是目前最流行的大中型关系数据库管理系统,从 SQL Server 7.0、SQL Server 2000、SQL Server 2005,发展到今天的 SQL Server 2008,其功能越来越强大。SQL Server 2008 可以为各类用户提供完善的数据库解决方案。

本书是一本详细讲解当前流行的 SQL Server 2008 数据库的项目化教程,全书分为 5 个项目共 15 章,项目 1 为数据库基本知识,包括 3 章,主要讲解数据库的基本知识、安装和配置 SQL Server 2008,以及 T-SQL 语言基础;项目 2 为创建和管理数据库,包括 4 章,主要讲解创建和管理数据库、数据表以及索引和数据的完整性;项目 3 为使用数据库,包括 5 章,主要介绍查询数据、视图、自定义函数和存储过程、触发器、游标及事务;项目 4 为保障数据库安全,包括 2 章,主要讲解了备份和恢复数据库、管理数据库安全;项目 5 为 SQL Server 2008 数据库应用实例,详细介绍了 SQL Server 2008 与 ASP.NET 联合开发学生成绩查询系统实例。

本书主要特色如下。

1. 编写过程中始终贯彻“面向高职高专学生,以训练基本技能为宗旨,以应用为目的,理论够用为度”的教学原则。本书以培养学生的应用能力为目的,通过实训环节加强实际应用能力的训练,在内容体系结构安排上有所创新。

2. 采用项目教学体系,项目作为主线贯穿整本教材的理论和实训内容。其中一个大项目贯穿于整本教材的理论内容,围绕该大项目,精心设计每一章的理论内容;另一个大项目贯穿于教材每一章后的实训项目。每一个实训项目的设计都力争做到既有针对性,又能够让学生通过项目举例很快掌握对应知识,并且所有示例都经过反复调试,保证其正确无误。

3. 内容和结构顺序设置合理,学生易学,教师易教。本教材从学生本位的角度考虑编写结构和组织内容,在语言的表述上力争通俗易懂,以便学生易于理解和学习。

4. 以职业能力为导向,培养学生的实际技能。在教材中,每章后附有“学生实训项目”,在其中设计了实训要求和实训思考题,并在教材的最后一章给出“SQL Server 2008 与 ASP.NET 联合开发学生成绩查询”项目。

本书编写人员长期从事 SQL Server 教学,并有一定的软件设计经验,并且在编写过程中多方听取了相关工程技术人员的意见。本书结构合理、知识精炼、注重项目实践、例题丰富、图文并茂、实用性强,不但可以作为大中专院校的数据库技术教材,也可以作为数据库技术初学者的入门指南。

本书由张宝华任主编,兰静、沈志梅任副主编。参加编写的还有吕梁高等专科学校的高勇强、白凤凤,潍坊职业学院的张长海、徐春华、徐希炜、朱丽兰、王英存、杨柳、武希英和吴磊,全书由张宝华统稿。

我们将为使用本书的教师免费提供电子教案,需要者可以到化学工业出版社教学资源网站 <http://www.cipedu.com.cn> 免费下载使用。

由于编者水平有限,书中不足之处在所难免,恳请广大读者批评指正。

编 者

2010 年 5 月

目 录

项目 1 数据库的基本知识

第 1 章 数据库的基础知识	1
1.1 数据库技术的产生与发展	1
1.2 数据库基本概念	3
1.2.1 基本概念	3
1.2.2 数据库三要素	4
1.2.3 数据库分类	4
1.3 数据库系统的体系结构	6
1.3.1 三级模式	6
1.3.2 数据库的两级映像功能	7
1.4 关系数据库	7
1.4.1 关系数据库的基本概念	7
1.4.2 关系的完整性	8
1.5 数据库系统设计	9
1.5.1 需求分析	9
1.5.2 概念结构设计	11
1.5.3 逻辑结构设计	13
1.5.4 物理结构设计	14
1.5.5 数据库的实施	14
1.5.6 数据库的运行和维护	15
1.6 项目举例	15
本章小结	16
1.7 理论知识练习题	17
1.8 实训项目一	18
第 2 章 SQL Server 2008 服务器的安装与配置	19
2.1 关系数据库语言 SQL	19
2.1.1 SQL 概述	19
2.1.2 SQL Server 的版本发展	19
2.1.3 SQL 的主要功能	20
2.1.4 SQL 的特点	20
2.2 SQL Server 2008 简介	21
2.2.1 SQL Server 2008 概述	21
2.2.2 SQL Server 2008 的新增功能	21
2.2.3 SQL Server 2008 的版本	24
2.3 安装 SQL Server 2008	25
2.3.1 安装 SQL Server 2008 的软、硬件要求	25
2.3.2 安装 SQL Server 2008 的步骤	26
2.4 SQL Server 2008 服务器组件	32
2.5 SQL Server 2008 管理和开发工具	32
2.5.1 SQL Server 2008 的管理工具	32
2.5.2 SQL Server Management Studio (SSMS) 环境	34
2.5.3 SQL 联机丛书	37
2.6 项目举例	38
本章小结	39
2.7 理论知识练习题	39
2.8 实训项目二	40
第 3 章 T-SQL 基础	41
3.1 SQL 与 T-SQL	41
3.2 数据类型	42
3.3 SQL 查询分析器的简单使用	46
3.3.1 SELECT 语句无源查询	46
3.3.2 简单 SELECT 语句解析	46
3.4 常量与变量	47
3.4.1 常量	47
3.4.2 变量	48
3.5 常用系统内置函数	49
3.5.1 数学函数	49
3.5.2 字符串函数	49
3.5.3 日期时间函数	51
3.5.4 聚合函数	51
3.5.5 系统函数	52
3.5.6 系统统计函数	52
3.5.7 游标函数	53
3.5.8 元数据函数	53
3.5.9 安全函数	53

3.6 运算符及表达式	54	3.7.2 流程控制语句	57
3.6.1 运算符及表达式	54	3.8 项目举例	59
3.6.2 运算符的优先级	55	本章小结	60
3.7 流程控制语句和批处理	56	3.9 理论知识练习题	61
3.7.1 批处理	56	3.10 实训项目三	62
项目 2 创建和管理数据库			
第 4 章 创建和管理数据库	63	6.1.2 域完整性	97
4.1 数据库的组成结构	63	6.1.3 参照完整性	97
4.1.1 系统数据库	63	6.1.4 用户定义完整性	98
4.1.2 数据库文件和文件组	64	6.2 使用约束	98
4.1.3 数据库对象	65	6.2.1 约束的定义	98
4.2 创建数据库	65	6.2.2 PRIMARY KEY 主键约束	98
4.2.1 使用图形化界面创建数据库	65	6.2.3 FOREIGN KEY 外键约束	99
4.2.2 用 T-SQL 语句创建数据库	67	6.2.4 DEFAULT 约束	100
4.3 管理数据库	69	6.2.5 UNIQUE 约束	101
4.3.1 用图形化界面管理数据库	69	6.2.6 CHECK 约束	102
4.3.2 用 T-SQL 语句管理数据库	71	6.2.7 删除约束	103
4.4 项目举例	74	6.3 使用 IDENTITY 列	103
本章小结	77	6.4 项目举例	105
4.5 理论知识练习题	77	本章小结	106
4.6 实训项目四	77	6.5 理论知识练习题	107
第 5 章 创建和管理数据表	79	6.6 实训项目六	107
5.1 数据表的组成结构	79	第 7 章 索引	109
5.2 创建数据表	79	7.1 索引概述	109
5.2.1 使用图形化界面创建数据表	79	7.1.1 索引的概念	109
5.2.2 用 T-SQL 语句创建数据表	81	7.1.2 创建索引的优缺点及使用原则	109
5.3 管理数据表	82	7.1.3 索引的类型	110
5.3.1 用企业管理器管理数据表	82	7.2 创建与管理索引	111
5.3.2 用 T-SQL 语句管理数据表	84	7.2.1 使用 T-SQL 语句创建与管理索引	111
5.4 管理数据表数据	85	7.2.2 使用图形化界面创建、查看、重命名及删除索引	113
5.4.1 用 SSMS 的对象资源管理器管理数据表数据	85	7.3 维护索引	113
5.4.2 用 T-SQL 语句操作数据表数据	86	7.3.1 显示索引的碎片信息	113
5.5 项目举例	89	7.3.2 整理碎片	114
本章小结	94	7.3.3 重建索引	114
5.6 理论知识练习题	94	7.4 项目举例	114
5.7 实训项目五	95	本章小结	115
第 6 章 保证数据的完整性	97	7.5 理论知识练习题	115
6.1 数据完整性的概念	97	7.6 实训项目七	116
6.1.1 实体完整性	97		

项目3 使用数据库

第8章 数据查询	118	9.5 删除视图	148
8.1 SELECT 语句概述	118	9.6 项目举例	148
8.2 T-SQL 简单查询	120	本章小结	150
8.2.1 使用 SELECT 子句查询字段和记录	120	9.7 理论知识练习题	150
8.2.2 使用 WHERE 子句进行条件查询	123	9.8 实训项目九	150
8.2.3 使用 INTO 子句将查询结果生成新表	125	第10章 自定义函数和存储过程	152
8.3 集合查询	125	10.1 自定义函数	152
8.3.1 并操作	125	10.1.1 自定义函数的概述	152
8.3.2 交操作	126	10.1.2 使用 SQL 语句创建、调用自定义函数	152
8.3.3 差操作	126	10.1.3 使用图形化界面创建自定义函数	155
8.4 统计查询	127	10.1.4 查看自定义函数	156
8.4.1 汇总查询(聚合函数)	127	10.1.5 修改自定义函数	156
8.4.2 使用分组	128	10.1.6 删除自定义函数	157
8.4.3 汇总计算	129	10.2 存储过程	157
8.5 连接查询	131	10.2.1 存储过程的概述	158
8.5.1 内连接(INNER JOIN)	132	10.2.2 使用 SQL 语句创建、执行存储过程	159
8.5.2 外连接(OUTER JOIN)	132	10.2.3 使用图形化界面创建存储过程	163
8.5.3 自连接(SELF JOIN)	134	10.2.4 查看存储过程	163
8.5.4 交叉连接(CROSS JOIN)	134	10.2.5 修改存储过程	165
8.6 嵌套查询	134	10.2.6 删除存储过程	166
8.6.1 带 IN 的嵌套查询	135	10.3 项目举例	166
8.6.2 带比较运算符的嵌套查询	136	本章小结	167
8.6.3 带 ANY 或 ALL 的嵌套查询	136	10.4 理论知识练习题	168
8.6.4 相关子查询	137	10.5 实训项目十	169
8.6.5 INSERT、DELETE 和 UPDATE 语句中的子查询	137	第11章 触发器	170
8.7 项目举例	138	11.1 触发器概述	170
本章小结	140	11.2 DML 触发器	171
8.8 理论知识练习题	140	11.2.1 DML 触发器的类型	171
8.9 实训项目八	141	11.2.2 DML 触发器的工作原理	171
第9章 视图	142	11.2.3 创建 DML 触发器	171
9.1 视图的概述	142	11.2.4 查看 DML 触发器	175
9.2 创建视图	143	11.2.5 禁用和启用 DML 触发器	175
9.3 使用视图	146	11.2.6 修改和删除 DML 触发器	176
9.3.1 查询视图	146	11.3 DDL 触发器	177
9.3.2 通过视图更新数据	146	11.3.1 创建 DDL 触发器	177
9.4 修改视图	147	11.3.2 查看、修改和删除 DDL 触发器	179
		11.4 项目举例	179
		本章小结	181

11.5 理论知识练习题	181	12.1.6 释放游标	189
11.6 实训项目十一	182	12.2 事务	189
第 12 章 游标及事务	183	12.2.1 事务的概念及特点	189
12.1 游标	183	12.2.2 事务的模式	190
12.1.1 游标的概念及特点	183	12.2.3 事务控制	192
12.1.2 声明游标	184	12.3 项目举例	193
12.1.3 打开游标	185	本章小结	195
12.1.4 使用游标处理数据	186	12.4 理论知识练习题	195
12.1.5 关闭游标	189	12.5 实训项目十二	196

项目 4 保障数据库安全

第 13 章 数据库的安全性	198	第 14 章 备份、恢复、分离、附加与导入	221
13.1 数据库安全概述	198	导出数据库	221
13.2 服务器的登录账户	200	14.1 备份概述	221
13.2.1 身份验证模式	200	14.1.1 备份的概念及类型	221
13.2.2 用企业管理器管理登录账户	201	14.1.2 备份设备	222
13.2.3 用 T-SQL 语句管理登录账户	202	14.1.3 备份的策略与规划	223
13.3 服务器角色	202	14.2 备份数据库	223
13.3.1 固定服务器角色	203	14.2.1 创建和管理磁盘备份设备	223
13.3.2 用企业管理器管理服务器角色	203	14.2.2 完全备份数据库	226
13.3.3 用 T-SQL 语句管理服务器角色	205	14.2.3 差异备份数据库	229
13.4 数据库用户	206	14.2.4 用事务日志备份数据库	230
13.4.1 用企业管理器管理数据库用户	207	14.2.5 用文件或文件组备份数据库	230
13.4.2 用 T-SQL 语句管理数据库用户	207	14.3 恢复数据库	231
13.5 数据库角色	208	14.3.1 恢复数据库的方式	232
13.5.1 固定数据库角色	208	14.3.2 用对象资源管理器恢复数据库	232
13.5.2 用企业管理器管理数据库角色	209	14.3.3 用 T-SQL 语句恢复数据库	233
13.5.3 用 T-SQL 语句管理数据库角色	210	14.4 分离与附加数据库	233
13.6 管理权限	210	14.4.1 分离数据库	233
13.6.1 权限概述	210	14.4.2 附加数据库	234
13.6.2 用企业管理器管理权限	211	14.5 数据导入与导出	235
13.6.3 用 T-SQL 语句管理权限	212	14.5.1 导入数据库	235
13.7 项目举例	215	14.5.2 导出数据库	241
本章小结	217	14.6 项目举例	244
13.8 理论知识练习题	217	本章小结	246
13.9 实训项目十三	219	14.7 理论知识练习题	247
		14.8 实训项目十四	248

项目 5 SQL Server 2008 数据库应用实例

第 15 章 SQL Server 2008 与 Asp.net 联合开发学生成绩查询系统	249	15.1.1 需求分析	249
15.1 系统设计	249	15.1.2 概念设计	250
		15.1.3 逻辑设计	251

15.1.4 功能设计	252	15.2.2 系统页面	255
15.2 程序设计	252	本章小结	258
15.2.1 数据库连接	252	15.3 实训项目十五	259
附录			
附录 A MyLibrary 数据库的表结构	260	附录 C 理论知识练习题参考答案	263
附录 B 物业信息管理库的表结构	261		
参考文献			268

项目 1 数据库的基本知识

第 1 章 数据库的基础知识

理论学习（教学）目标：

1. 了解数据库的产生与发展。
2. 熟悉数据库的体系结构。
3. 掌握数据库的基本概念。
4. 掌握关系数据库的基本概念。

技能学习（教学）目标：

1. 掌握数据库的概念结构的设计方法。
2. 掌握关系数据库的逻辑结构的设计方法。
3. 掌握数据库应用系统的设计过程和设计方法。

数据库技术就是研究如何科学地管理数据以便为人们提供可共享的、安全的、可靠的数据的技术。信息需求的增长使数据库的应用日益重要，范围日益广泛，数据库管理系统正逐渐应用到前所未有的应用领域。目前，数据库系统已经应用到医学、计算机辅助设计、能源管理、航空系统、天气预报、交通、旅馆、资料以及人力资源管理等领域。数据库系统的发展满足了用户共享信息的需求。随着在线信息的增加以及越来越多的用户希望访问在线信息，今后还会开发出更多的数据库系统。

1.1 数据库技术的产生与发展

使用计算机后，数据处理的速度及规模都是过去采用人工或机械方式无法比拟的。随着数据处理量的不断增加，数据管理技术也应运而生，其演变过程随着计算机硬件和软件的发展不断变化。学校对学生信息的管理方式的演变就是最好的实例。在没有使用计算机时，学生管理工作人员将学生的个人信息以卡片形式存放。为了便于查找，将同一个班级、同一个年级、同一个系的学生卡片相邻存放，并对不同的班级、年级、系分别做上标签。学校还以同样的方式管理学生的学习成绩，在学期末将同一个班的学生成绩单装入档案袋收藏起来。如果要查找一个学生的信息，就可以按照标签迅速找到。如果需要计算某个学生的平均成绩就要在档案袋中找到该学生的所有课程的成绩单，再计算平均值。在计算机高速发展的今天，大部分学校已经可以通过学生成绩管理系统来快速查找学生的个人信息，并且可以准确地查看到学生的所有成绩了。

总的来说,数据库技术的发展经历了以下几个阶段。

1. 人工管理阶段

计算机没有应用到数据管理领域之前,数据管理的工作是由人工完成的。

20 世纪 50 年代中期以前,计算机主要用于科学计算。当时的硬件状况是,外存只是纸带、卡片或磁带,没有磁盘等直接存取的存储设备;软件状况是,没有操作系统,没有管理数据的软件,因此称这一阶段的数据管理方式为人工管理数据。人工管理数据具有如下特点。

(1) 数据不便保存。

(2) 应用程序管理数据。数据需要应用程序自行管理,没有相应的软件系统负责数据的管理工作。

(3) 数据不共享。数据是面向应用的,一组数据只对应一个应用程序。当多个应用程序涉及某些相同的数据时,必须各自定义,无法相互利用、相互参照。

(4) 数据不具有独立性。数据的逻辑结构或物理结构改变后,必须对应用程序做相应的修改。

2. 文件系统阶段

20 世纪 50 年代后期到 60 年代中期,在计算机硬件方面已有了磁盘、磁鼓等直接存储设备;软件方面,操作系统中已经有了专门的数据管理软件——文件系统。可以把相关的数据组织成一个文件存放在计算机中,需要时只要提供文件名,计算机就能从文件系统中找出所要的文件,并把文件中存储的数据提供给用户进行处理。

使用文件系统管理数据具有如下特点。

(1) 数据可以长期保存。数据可以组织成文件长期保存在计算机中并反复使用。

(2) 由文件系统管理数据。文件系统把数据组织成内部有结构的记录,实现“文件名访问,按记录进行存取”的管理技术。

(3) 文件系统使应用程序与数据之间有了初步的独立性,程序员不必过多地考虑数据存储的物理细节。

使用文件系统管理数据具有以下缺点。

(1) 数据共享性差,冗余度大。在文件系统中,一个(或一组)文件基本上对于一个应用(程序),当不同的应用程序使用相同的数据时,必须建立各自的文件,而不能共享相同的数据。因此数据的冗余度大,浪费存储空间。

(2) 数据独立性差。对现有的数据再增加一些新的应用很困难,系统不容易扩充。一旦数据的逻辑结构发生改变,就必须修改应用程序,修改文件结构的定义。

3. 数据库系统阶段

20 世纪 60 年代后期,计算机在数据管理上的应用越来越广泛,数据量急剧增长,同时对多种应用、多种语言互相覆盖的共享数据集合的需求也越来越强烈。

这时已有大容量磁盘,硬件的价格下降;软件的价格则上升,为编制和维护系统软件及应用程序所需的成本相对增加。在这种背景下,以文件系统作为数据管理的手段已不能满足应用的需求。于是,为解决多用户、多应用共享数据的要求,使数据为尽可能多的应用服务,数据库技术应运而生,出现了统一管理数据的专用软件系统——数据库管理系统。

用数据库管理系统来管理数据和使用文件系统相比具有明显的优点,从文件系统到数据库管理系统,标志着数据管理技术的飞跃。由于数据库是以数据为中心组织数据的,减少了数据的冗余,提供了更高的数据共享能力,同时要求程序和数据具有较高的独立性,因此当数据的逻辑结构改变时,不涉及数据的物理结构,也不影响应用程序,这样就降低了程序研制与维护的费用。

4. 高级数据库阶段

这一阶段的主要标志是 20 世纪 80 年代的分布式数据库系统、90 年代的对象数据库系统和

21 世纪初的网络数据库系统的出现。

(1) 分布式数据库系统。在这一阶段以前, 数据库系统是集中式的。集中式数据库把数据库集中在一个数据库中进行管理, 不但减少了数据冗余和不一致性; 而且数据联系更强。但集中式数据库也有弱点: 一是随着数据量的增加, 系统会变相当庞大, 操作复杂, 开销大; 二是数据集中存储, 大量的通信都要通过主机, 造成拥挤。随着小型计算机和微型计算机的普及, 计算机网络软件和远程通信技术的发展, 分布式数据库系统崛起了。

分布式数据库系统主要有以下 3 个特点。

① 数据库的数据在物理上可分布在不同的地方, 但逻辑上是一个整体。

② 各个分散的数据库既可以执行局部应用 (访问本地数据库), 又可以执行全部应用 (访问异地数据库)。

③ 各个分散的计算机由数据通信网络连接。本地计算机不能单独胜任的处理任务, 可以通过网络取得其他数据库和计算机的支持。

分布式数据库系统兼顾了集中管理和分布处理两个方面, 因而有良好的性能。

(2) 对象数据库系统。在数据处理领域, 关系数据库的应用已相当普遍、相当出色。但是现实世界存在着许多具有更复杂数据结构的实际应用领域, 如多媒体数据、多维表格数据、CAD 数据等, 已有的层次、网状、关系三种数据模型对这些应用领域都显得力不从心。对象数据库正是由于适应这种形式而发展起来的, 它是面向对象的程序设计技术与数据库技术结合的产物。

对象数据库系统主要有以下 2 个特点。

① 对象数据库模型能完整地描述现实世界的数据结构, 能表达数据间嵌套、递归等关系。

② 具有面向对象技术的封装性 (把数据与操作定义在一起) 和继承性 (继承数据结构和操作) 的特点, 提高了软件的可重用性。

(3) 网络数据库系统。C/S (客户机/服务器) 结构的出现, 使得人们可以更有效地使用计算机资源。但在网络环境中, 如何隐藏各种复杂性, 这就要使用中间件。其中涉及数据访问的中间件, 就是 20 世纪 90 年代提出的 ODBC 和 JDBC 技术。

现在, 信息已经成为各行各业的宝贵财富, 计算机网络已成为信息化社会中十分重要的基础设施。随着广域网 (WAN) 的发展, 智能化、高性能的数据库系统与功能强大的计算机相结合, 用于实现通信交往、资源共享或协调工作等目标, 正在对社会的发展起着巨大的推进作用。

1.2 数据库基本概念

1.2.1 基本概念

1. 数据 (Data)

数据是数据库中存储的基本对象, 是描述事物的符号。广义上讲, 数据可以是数字、文字、图形、图像、声音和语言等, 即数据有多种形式, 但都是经过数字化存入计算机的。

例如: 郑超, 男, 1989 年 5 月 25 日出生, 信息工程系 090301 班的学生。

2. 数据库 (Data Base, DB)

数据库 (DB) 是存放数据的仓库, 这些数据间存在一定的关联, 在计算机上需要有存储空间和一定的存储格式。严格地讲, 数据库是被长期存放在计算机内的、有组织的、统一管理的相关数据的集合, 能为用户共享, 具有最小冗余度, 数据间联系密切, 有较高的独立性。

例如, 学校为了便于管理, 把学生个人信息、课程、学生成绩、教师信息等数据有序地存放在计算机内, 就可以构成一个数据库。

3. 数据库管理系统 (Data Base Management System, DBMS)

建立了数据库之后,下一个问题就是如何科学地组织和存储数据,如何高效地获取和维护数据。完成这个任务的就是——数据库管理系统。

数据库管理系统(DBMS)是对数据进行管理的软件,位于用户与操作系统之间,属于系统软件,它为用户或应用程序提供访问数据库的方法。包括数据库的建立、查询、更新及各种数据控制方法。

4. 数据库管理员 (Data Base Administrator, DBA)

数据库的建立、使用和维护等工作只靠一个 DBMS 是不可能完成的,还要有专门的人员来完成,这些人被称为数据库管理员(DBA)。数据库管理员负责管理和监控数据库系统,负责为用户解决在应用中出现的系统问题。

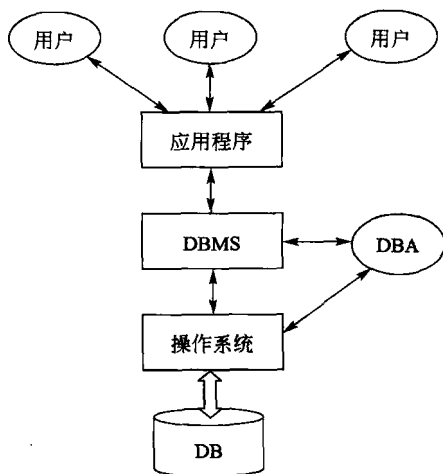


图 1.1 数据库系统简图

5. 数据库系统 (Data Base System, DBS)

数据库系统(DBS)是指在计算机系统中引入数据库后的系统。一个完整的数据库系统由数据、数据库、数据库管理系统、操作数据库的应用程序、数据库管理员以及支撑的硬件平台和软件平台构成。

如图 1.1 所示为数据库系统构成简图(其中硬件、系统软件没有画出来)。

为什么要使用数据库系统呢?这是因为数据库系统为数据提供了共享、稳定、安全的保障体系。如果用户需要持久地存储数据,数据库无疑是维护这些持久数据的最合适的地方;如果用户管理的数据具有结构性强、相互之间有联系、数据的取值有约束等特征,为了管理方便,也应该使用数据库系统;同时数据库系统提供了功能强大的数据查询功能。例如在图书信息管理中,

管理图书、借阅人以及借阅情况就适合使用数据库系统来完成。

1.2.2 数据库三要素

模型是对现实世界的抽象,如一张地图、一架航模飞机等。在数据库技术中人们用数据模型来描述数据库的结构和语义,对现实世界进行抽象,在这里模型描述的是事物的表征及特征。数据库的数据模型应包含数据结构、数据操作、完整性约束 3 个要素。

1. 数据结构

数据结构用于描述数据库的静态特性,是所研究的对象类型的集合(数据定义)。是对实体类型和实体间联系的表达和实现。

2. 数据操作

数据操作用于描述数据库的动态特性,是指对数据库中各种对象的实例允许执行的操作的集合(如查询、插入、更新、删除等)。

3. 完整性约束

数据的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及其联系所具有的制约和存储规则,用以限定数据库的状态以及状态的变化,保证数据的正确性、有效性和相容性。

1.2.3 数据库分类

数据库管理系统根据数据模型对数据进行存储和管理,数据库管理系统采用的数据模型主要

有层次模型、网状模型和关系模型。以数据模型为主线，数据库主要分为层次数据库、网状数据库和关系数据库。其中层次数据库和网状数据库统称为非关系数据库。

1. 层次数据库

层次模型是数据库系统中最早出现的数据模型，用树形层次结构表示各类实体以及实体间的联系。层次模型数据库的典型代表是 IBM 公司的数据库管理系统（Information Management Systems, IMS），这是一个最早推出的数据库管理系统。

如图 1.2 所示为某学校按层次模型组织的数据示例。

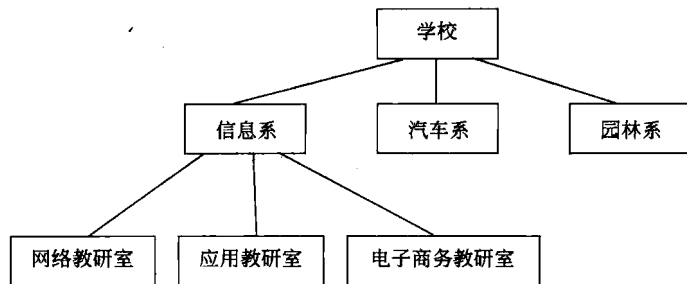


图 1.2 简单的层次模型

层次模型对具有一对多的层次关系的描述非常自然、直观、容易理解，这是层次数据库的突出优点。

2. 网状数据库

网状数据模型的典型代表是 DBTG 系统，也称 CODASYL 系统，它是 20 世纪 70 年代数据系统语言研究会（Conference On Data Systems Language, CODASYL）下属的数据库任务组（Data Base Task Group, DBTG）提出的一个数据模型方案。

网状模型中的每一个数据用一个节点表示，每个节点与其他节点都有联系，这样，数据库中的所有节点就构成了一个复杂的网络。若用图表示，网状模型是一个网络。如图 1.3 所示为一个抽象的简单的网状模型。

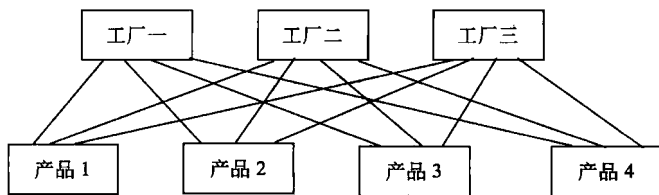


图 1.3 简单的网状模型

3. 关系数据库

关系模型是目前应用最广泛的一种数据模型。IBM 公司的研究员 E. F. Codd 于 1970 年发表了题为“大型共享系统的关系数据库的关系模型”的论文，文中首次提出了数据库系统的关系模型。20 世纪 80 年代以来，计算机厂商新推出的数据库管理系统（DBMS）几乎都支持关系模型，非关系模型的产品也大都加上了关系接口。当前数据库领域的研究工作都是以关系模型为基础的。

关系模型用二维表格结构表示实体集，用键来表示实体间联系。这个二维表在关系数据库中就称为关系。

例如，在学生成绩管理系统中，学生个人信息表中，涉及的主要信息有学号、姓名、性别、出生日期、民族、籍贯等。表 1.1 描述了学生成绩管理系统中部分学生的数据信息。

表 1.1 学生个人信息表

学 号	姓 名	性 别	民 族	出 生 日 期	籍 贯
0906001	郑超	男	汉族	1990-01-27	山东省五莲县
0906002	韩巨东	男	汉族	1989-06-16	山东省寿光市
0906003	孙野	女	汉族	1988-04-16	山东省莱阳市
0906004	徐小舟	男	汉族	1989-01-13	山东省莒南县
0906005	陈孝庭	男	汉族	1990-07-27	河北省平山县
0906006	席光楠	女	汉族	1990-10-15	山东省安丘市
0906007	马起俊	男	汉族	1989-04-16	山东省东明县
0906008	杨亚南	女	汉族	1990-08-15	山东省安丘市

1.3 数据库系统的体系结构

1.3.1 三级模式

从数据库管理系统角度看，数据库系统的体系结构分成三级：内模式（内部级）、概念模式（概念级）、外模式（外部级），即三级模式结构。如图 1.4 所示学生信息数据库系统的三级模式结构。

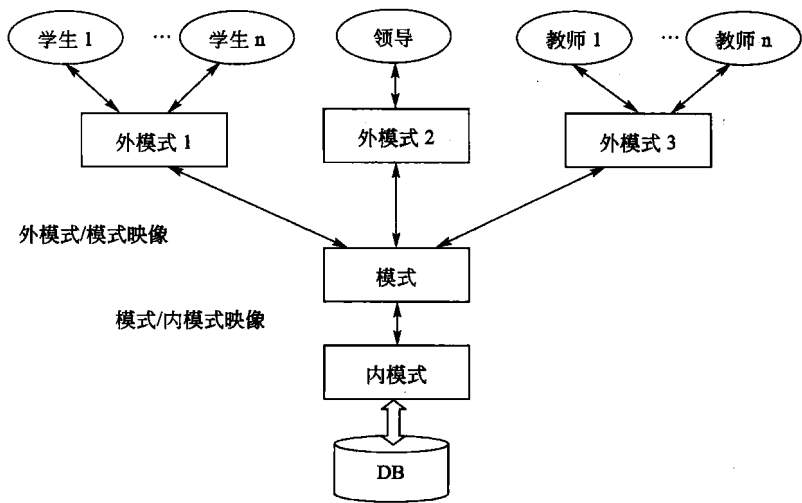


图 1.4 学生信息数据库系统的三级模式结构

(1) 外模式也称用户模式或子模式，是从数据库用户的角度看到的数据结构的描述，是用户与数据库系统的接口，是数据库用户的数据视图。同一类用户使用同一个外模式，是保证数据库安全的一个措施。

外模式通常是模式的子集。一个模式可以有多个外模式，同一个外模式也可以为某一用户的多个应用系统使用，但一个应用程序只能使用一个外模式。外模式是保证数据库安全的一个有力措施。每个用户只能看到和访问所对应的外模式中的数据，数据库中的其他数据是看不到的。

(2) 概念模式也称模式，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。这种描述是一种抽象描述，不涉及具体的硬件环境与平台，也与具体的软件环境无关。

一个数据库只有一个模式，在定义数据时应首先定义模式，即定义数据的逻辑结构（如数据

项、名字、类型等)和数据之间的联系。模式的一个具体值称为模式的一个实例。模式是相对稳定的,而实例是相对变动的,因为数据库中的数据是在不断更新的。

(3)内模式也称存储模式,是数据物理结构和存储方式的描述,一个数据库只有一个内模式。

1.3.2 数据库的两级映像功能

数据库系统的内模式、模式、外模式之间有很大的差别。为了实现用户和数据之间的透明化,DBMS在这三级模式之间提供了两层映像:外模式/模式映像和模式/内模式映像。

正是这两层映像保证了数据库系统的数据能够具有较高的逻辑独立性和物理独立性。

映像实质上是一种对应关系,是指映像双方如何进行数据转换,并定义转换规则。这样就能使数据的独立性得到保证。

1. 外模式/模式映像

数据库中的每一个外模式都有一个外模式/模式映像,定义了该外模式与模式之间的对应关系。如果模式改变,则需要对各个外模式/模式映像做相应的改变,从而使外模式保持不变,而不必修改外模式的应用程序,保证了数据与应用程序的逻辑独立性。

2. 模式/内模式映像

数据库中的模式/内模式映像 is 唯一的,因为数据库只有一个模式和一个内模式。模式/内模式映像定义了数据库的逻辑结构与存储结构之间的对应关系,如果数据库的存储结构改变,则对模式/内模式映像做相应的改变,使模式保持不变,从而不必修改模式的应用程序,保证了数据与应用程序的物理独立性。

为了便于理解三级模式的概念,可用图书馆做个比喻。图书馆中的书库是存放各类图书的仓库。这些图书按照类别摆在书架上,相当于数据库的内模式(存储模式)。为了借阅方便,需要编制一套书目卡片,书目卡片与书架上的书一一对应,书目卡片就相当于模式。书目卡片与书架的对应关系就相当于模式/内模式映像。读者不需要知道全部图书的具体存放位置,只需要知道所要借阅图书的书目卡片(模式)的一部分(外模式),就可以借到所需图书。图书的存放位置改变了,不会影响到读者按照书目卡片借书,而书库中的书是供所有读者共享的。读者通过图书管理员可以借到所需要的图书,图书管理员就相当于数据库管理系统。

1.4 关系数据库

关系数据库是目前数据库应用中的主流技术。关系数据库之所以得到广泛应用,是因为它是建立在严格的数学理论基础上的,概念清晰、简单,能够用统一的结构来表示实体集合和它们之间的联系。从数据库的发展历程中可以看到,关系数据库的出现标志着数据库技术走向成熟。关系数据库的标准语言是SQL,将在第2章中介绍。

1.4.1 关系数据库的基本概念

在关系数据库中,用单一的二维表结构来表示实体及实体间的关系,如图1.5所示。

(1)关系。一个关系对应一个二维表,二维表的表名就是关系名。图1.5中包含两个二维表,即两个关系:学生信息关系和课程成绩关系。

(2)属性和值域。二维表中的每一列都称为关系的属性。属性包括属性名和属性值两部分,列名即为属性名,列值即为属性值。属性的取值范围称为值域。

在图1.5中,学生个人信息关系中有学号、姓名、性别、出生日期4个属性,是四元关系。其中,如性别的值域是{'男','女'},成绩的值域是0~100。

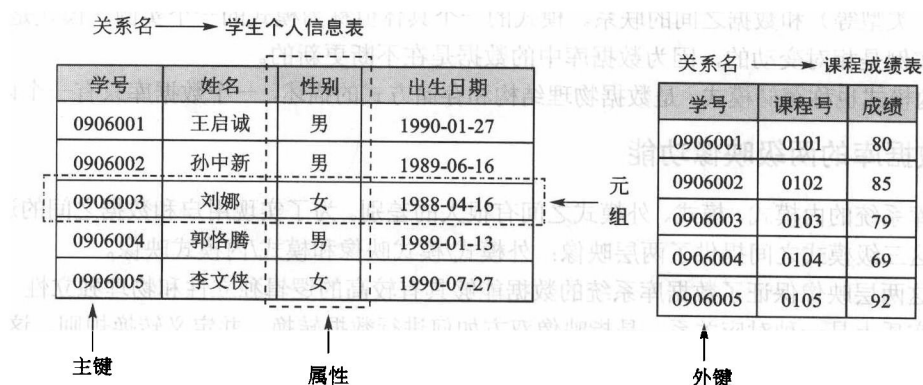


图 1.5 关系数据库的基本概念

(3) 关系模式。二维表中的行定义（表头），即对关系的描述称为关系模式，关系模式的一般形式为

关系名（属性 1，属性 2，…，属性 n ）

图 1.5 中的两个关系模式为

学生个人信息关系（学号，姓名，性别，出生日期）

课程成绩关系（学号，课程号，成绩）

(4) 元组。表中的每行数据称为一个元组，也称为一条记录。如（0906003，刘娜，女，1988-04-16）、（0906004，0104，69）

(5) 主键（Primary Key）。主键也称主关键字或主码，是表中的属性或属性的组合，用于确定唯一的一个元组。主键不能是空值。如图 1.5 所示，学号是学生个人信息表中的主键，（学号，课程号）是课程成绩表的主键。

(6) 外键（Foreign Key）。外键的定义是相对于主键而言的，是用于建立和加强两个表数据之间的链接的一列或多列。

如图 1.5 所示，学号是学生个人信息表中的主键，而课程成绩表中也出现了学号，则相对于学生个人信息表，学号就是课程成绩表的外键。

1.4.2 关系的完整性

关系的完整性是为保证数据库中数据的正确性和相容性，对关系模型提出的某种约束条件或规则。完整性通常包括实体完整性、域完整性、参照完整性和用户定义完整性，其中实体完整性和参照完整性是关系模型必须满足的完整性约束条件。

1. 实体完整性

实体完整性是指关系的主键不能取“空值”。一个关系对应现实世界中一个实体集，如图 1.5 所示关系就对应学生这一实体集。现实世界中的实体是可相互区分、识别的，也即它们应具有某种唯一性的标识。在关系模式中，以主键作唯一性标识，而主键中的属性（称为主属性）不能取空值，否则，表明关系模式中存在着不可标识的实体（因空值是“不确定”的），这与现实世界的实际情况相矛盾，这样的实体就不是一个完整实体。按实体完整性规则要求，主属性不能取空值，如果主键是多个属性的组合，则所有主属性均不得取空值。

图 1.5 中“学号”列作为学生个人信息表的主键，该列不得有空值，否则无法对应某个具体的学生，这样的表格不完整，对应关系不符合实体完整性规则的约束条件。（学号，课程号）是