高等院校计算机应用系列教材

MySQL 数据库原理与应用

(微课版)

张星秋 编著

清華大學出版社

北 京

内 容 简 介

本书较全面地介绍了 MySQL 数据库的基础知识及其应用。全书共分为 17 章，主要内容包括数据库基础、初始 MySQL、MySQL 图形化管理工具、数据库操作、数据表操作、数据记录操作、数据查询、MySQL 函数、运算符、视图、存储程序、触发器、MySQL 权限与安全管理、数据备份与恢复、MySQL 日志、性能优化等。本书采用理论与实践相结合的方式，每章结合示例来阐述知识要点，章末进行本章小结，并辅以思考与练习巩固所学。

本书内容丰富、结构合理、思路清晰、语言简练流畅、示例翔实，既可作为高等院校数据库基础或数据库开发课程的教材，也可作为计算机软件开发人员、从事数据库管理与维护工作的专业人员、广大计算机爱好者的自学用书。

本书配套的电子课件、实例源文件、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获取。扫描前言中的视频二维码可以直接观看教学视频。

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。
版权所有，侵权必究。举报：010-62782989，beiqinquan@tup.tsinghua.edu.cn。

图书在版编目（CIP）数据

MySQL 数据库原理与应用 : 微课版 / 张星秋编著. 北京 : 清华大学出版社, 2024. 7. -- (高等院校计算机应用系列教材). -- ISBN 978-7-302-66582-3

Ⅰ. TP311.132.3

中国国家版本馆 CIP 数据核字第 2024VD3411 号

责任编辑：胡辰浩
封面设计：高娟妮
版式设计：芃博文化
责任校对：孔祥亮
责任印制：丛怀宇

出版发行：清华大学出版社
网　　址：https://www.tup.com.cn，https://www.wqxuetang.com
地　　址：北京清华大学学研大厦 A 座　　邮　　编：100084
社 总 机：010-83470000　　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：河北鹏润印刷有限公司
经　　销：全国新华书店
开　　本：185mm×260mm　　**印　　张：**24　　**字　　数：**677 千字
版　　次：2024 年 9 月第 1 版　　**印　　次：**2024 年 9 月第 1 次印刷
定　　价：79.00 元

产品编号：101297-01

前　言

本书从一个新手的视角出发介绍 MySQL 8.0 数据库管理系统。MySQL 是一款非常优秀的自由软件，而且已经是世界上最流行的数据库之一。国内很多大型的企业都选择 MySQL 作为数据库，对 MySQL 数据库技术人员的需求旺盛，很多知名企业都在招聘技术能力强的 MySQL 数据库技术人员和管理人员，这些都证明了 MySQL 数据库的可靠性、实用性和受欢迎程度。

本书以 MySQL 8.0 版本为基础，针对初学者量身定做，内容注重实战，通过实例的操作与分析，引领读者快速学习和掌握 MySQL 开发和管理技术。全书内容分 17 章，各章主要内容如下。

第 1 章主要介绍数据库基础知识，包括数据库技术概述、数据库的体系结构、E-R 图和数据库设计过程与规范。

第 2 章主要介绍 MySQL 数据库的发展史、优势、应用环境和 MySQL 8.0 新特性，在 Windows 10 平台和 Linux 环境下安装和配置 MySQL，启动和登录 MySQL 数据库，以及如何学好 MySQL。

第 3 章介绍 MySQL 图形化管理工具，重点介绍 phpMyAdmin 和 Navicat 两款工具的使用。

第 4 章介绍 MySQL 数据库的基本操作，包括创建数据库、删除数据库和 MySQL 数据库存储引擎等内容。

第 5 章介绍 MySQL 数据表的基本操作，包括创建数据表、查看数据表结构、修改数据表和删除数据表等内容。

第 6 章介绍数据记录的基本操作，包括插入、修改、删除数据记录等操作。

第 7 章介绍如何查询数据表中的数据，包括基本查询语句、按条件查询、高级查询、聚合函数查询、连接查询、子查询、合并查询结果、定义表和字段的别名、使用正则表达式进行查询等内容。

第 8 章介绍 MySQL 函数，包括数学函数、字符串函数、日期和时间函数、条件判断函数、系统信息函数、加密函数和其他函数。

第 9 章介绍 MySQL 运算符，包括运算符介绍、算术运算符、比较运算符、逻辑运算符、位运算符、运算符的优先级及运算符示例等内容。

第 10 章介绍 MySQL 视图，包括视图的概念、创建视图、查看视图、修改视图、更新视图和删除视图等内容。

第 11 章介绍 MySQL 中的存储过程和函数，包括存储过程和函数的创建、调用、查看、修改和删除等内容。

第 12 章介绍 MySQL 触发器，包括创建触发器、查看触发器、触发器的使用和删除触发器等内容。

第 13 章介绍 MySQL 权限与安全管理，包括 MySQL 中的各种权限表、账户管理、权限管理和 MySQL 的访问控制机制等内容。

第 14 章介绍 MySQL 数据库的备份和恢复，包括数据备份、数据恢复、数据库迁移、数据表的导出和导入等内容。

第 15 章介绍 MySQL 日志，包括日志简介、二进制日志、错误日志、通用查询日志和慢查询日志等内容。

第 16 章介绍如何对 MySQL 进行性能优化，包括优化简介、优化查询、优化数据库结构和优化 MySQL 服务器等内容。

第 17 章介绍图书管理系统数据库的设计方法和实现过程，以及网上购物商城系统的实现。

本书内容丰富、结构合理、思路清晰、语言简练流畅、示例翔实。每一章的引言部分概述了本章的知识内容和学习目标。在每一章的正文中，结合所讲述的关键知识点，穿插了大量极富实用价值的示例。每一章末尾都安排了本章小结和有针对性的思考和练习，有助于读者巩固本章所学的基本概念和培养实际动手能力。

本书既可作为高等院校数据库基础或数据库开发课程的教材，也可作为计算机软件开发人员、从事数据库管理与维护工作的专业人员、广大计算机爱好者的自学用书。

本书由张星秋编写。在编写本书的过程中参考了相关文献，在此向这些文献的作者深表感谢。由于作者水平有限，书中难免有欠妥之处，恳请专家和广大读者批评指正。我们的电话是 010-62796045，邮箱是 992116@qq.com。

本书配套的电子课件、实例源文件、习题答案可以到 http://www.tupwk.com.cn/downpage 网站下载，也可以扫描下方左侧的二维码获取。扫描下方右侧的二维码可以直接观看教学视频。

扫描下载

配套资源

扫一扫

看视频

编　者

2024 年 4 月于武汉

目　录

第 1 章　数据库基础……1
1.1　数据库概述……1
1.1.1　数据库技术的发展……1
1.1.2　数据库系统的组成……2
1.1.3　数据模型与规范化……2
1.1.4　结构化查询语言(SQL)……7
1.2　数据库的体系结构……7
1.2.1　数据库三级模式……8
1.2.2　三级模式之间的映射……8
1.3　E-R图……9
1.3.1　实体和属性……9
1.3.2　关系……10
1.3.3　E-R图设计原则……10
1.4　数据库设计……11
1.4.1　为实体建立数据表……11
1.4.2　为表建立主键和外键……11
1.4.3　为多对多关系建立数据表……12
1.4.4　为字段选择合适的数据类型……12
1.4.5　定义约束条件……13
1.5　本章小结……13
1.6　思考与练习……14

第 2 章　初识 MySQL……15
2.1　MySQL概述……15
2.1.1　MySQL的发展史……15
2.1.2　MySQL的优势……16
2.1.3　MySQL的应用环境……16
2.1.4　MySQL 8.0的新特性……17
2.2　Windows平台下安装与配置MySQL……18
2.2.1　MySQL服务器安装包的下载……18
2.2.2　MySQL服务器的安装……20
2.3　启动服务器并登录MySQL服务器……28
2.3.1　配置Path变量……28
2.3.2　启动和停止MySQL……30
2.3.3　连接和断开MySQL……31
2.3.4　打开MySQL 8.0 Command Line Client……32
2.4　Linux平台下安装和配置MySQL……32
2.4.1　下载并安装MySQL……33
2.4.2　通过apt安装MySQL服务……34
2.5　如何学好MySQL……35
2.6　本章小结……36
2.7　思考与练习……36

第 3 章　MySQL 图形化管理工具……37
3.1　MySQL图形化管理工具概述……37
3.2　phpMyAdmin……37
3.2.1　phpMyAdmin简介……37
3.2.2　安装phpStudy……38
3.2.3　下载phpMyAdmin……38
3.2.4　打开phpMyAdmin……39
3.2.5　数据库操作管理……40
3.2.6　管理数据表……42
3.2.7　管理数据记录……44
3.2.8　导入/导出数据……48
3.2.9　设置编码格式……49
3.2.10　添加服务器用户……50
3.2.11　重置MySQL服务器登录密码……52

3.3 Navicat……53
3.3.1 下载Navicat……53
3.3.2 安装Navicat……54
3.3.3 服务器连接……56
3.3.4 创建数据库……57
3.3.5 新建数据表……58
3.3.6 添加数据记录……60
3.3.7 导出/导入数据……60
3.3.8 “工具”菜单……62
3.4 本章小结……70
3.5 思考与练习……70

第 4 章 数据库操作……71
4.1 关系数据库简介……71
4.1.1 关系数据库基础知识……71
4.1.2 数据库常用对象……72
4.1.3 系统数据库……72
4.2 操作数据库……73
4.2.1 创建数据库……73
4.2.2 查看数据库……76
4.2.3 选择数据库……77
4.2.4 修改数据库……77
4.2.5 删除数据库……78
4.3 存储引擎……79
4.3.1 MySQL存储引擎的概念……79
4.3.2 MySQL支持的存储引擎……80
4.3.3 InnoDB存储引擎……81
4.3.4 MyISAM存储引擎……82
4.3.5 MEMORY存储引擎……83
4.3.6 如何选择存储引擎……84
4.3.7 设置存储引擎……84
4.4 本章小结……86
4.5 思考与练习……86

第 5 章 数据表操作……87
5.1 数据表基本操作……87
5.1.1 创建数据表……87
5.1.2 查看表结构……89
5.1.3 复制数据表……90
5.1.4 修改表结构……92
5.1.5 重命名数据表……95
5.1.6 删除数据表……95
5.2 数据类型……96
5.2.1 数字类型……96
5.2.2 字符串类型……97
5.2.3 日期和时间类型……98
5.2.4 如何选择数据类型……98
5.3 表约束操作……100
5.3.1 设置表字段的非空约束……100
5.3.2 设置表字段的默认值……101
5.3.3 设置表字段的唯一约束……102
5.3.4 设置表字段的主键约束……103
5.3.5 设置表字段值自动增加……105
5.3.6 设置表字段的外键约束……106
5.4 索引操作……108
5.4.1 索引概述……108
5.4.2 创建索引……109
5.4.3 删除索引……111
5.5 本章实战……112
5.6 本章小结……116
5.7 思考与练习……116

第 6 章 数据记录操作……117
6.1 插入数据记录……117
6.1.1 使用INSERT...VALUES语句插入单条记录……117
6.1.2 使用INSERT...VALUES语句插入多条记录……120
6.1.3 使用INSERT...SELECT语句插入结果集……120
6.1.4 使用REPLACE语句插入新数据记录……122
6.2 修改数据记录……123
6.3 删除表记录……124
6.3.1 使用DELETE语句删除表记录……124
6.3.2 使用TRUNCATE语句清空表记录……125
6.4 本章实战……125
6.5 本章小结……129

6.6 思考与练习……129

第 7 章 数据查询……130
7.1 基本查询……130
7.1.1 SELECT语句……130
7.1.2 查询所有字段……132
7.1.3 查询指定字段……133
7.1.4 查询指定数据……133
7.2 按条件查询……134
7.2.1 带关系运算符的查询……134
7.2.2 带IN关键字的查询……134
7.2.3 带BETWEEN AND关键字的查询…135
7.2.4 空值查询……135
7.2.5 用关键字DISTINCT去除结果中的重复行……137
7.2.6 带LIKE关键字的查询……138
7.2.7 带AND关键字的多条件查询……138
7.2.8 带OR关键字的多条件查询……139
7.3 高级查询……140
7.3.1 对查询结果排序……140
7.3.2 分组查询……140
7.3.3 使用LIMIT限制查询结果数量……142
7.4 聚合函数查询……142
7.4.1 COUNT函数……142
7.4.2 SUM函数……143
7.4.3 AVG函数……143
7.4.4 MAX函数……144
7.4.5 MIN函数……144
7.5 连接查询……144
7.5.1 内连接查询……144
7.5.2 外连接查询……146
7.5.3 复合条件连接查询……148
7.6 子查询……149
7.6.1 带IN关键字的子查询……149
7.6.2 带比较运算符的子查询……150
7.6.3 带EXISTS关键字的子查询……150
7.6.4 带ANY关键字的子查询……151
7.6.5 带ALL关键字的子查询……152
7.7 合并查询结果……153
7.7.1 使用UNION关键字……153
7.7.2 使用UNION ALL关键字……154
7.8 定义表和字段的别名……154
7.8.1 为表取别名……154
7.8.2 为字段取别名……155
7.9 使用正则表达式查询……155
7.10 本章小结……158
7.11 思考与练习……158

第 8 章 MySQL 函数……159
8.1 MySQL函数简介……159
8.2 数学函数……159
8.3 字符串函数……161
8.4 日期和时间函数……169
8.4.1 获取当前日期的函数和获取当前时间的函数……169
8.4.2 获取当前日期和时间的函数……169
8.4.3 UNIX时间戳函数……170
8.4.4 返回UTC日期的函数和返回UTC时间的函数……171
8.4.5 获取月份的函数……171
8.4.6 获取星期的函数……171
8.4.7 获取星期数的函数……172
8.4.8 获取天数的函数……173
8.4.9 获取年份、季度、小时、分钟和秒钟的函数……174
8.4.10 获取日期的指定值的函数……175
8.4.11 时间和秒钟转换的函数……175
8.4.12 计算日期和时间的函数……176
8.4.13 将日期和时间格式化的函数……178
8.5 条件判断函数……181
8.5.1 IF(expr,v1,v2)函数……181
8.5.2 IFNULL(v1,v2)函数……182
8.5.3 CASE函数……182
8.6 系统信息函数……183
8.6.1 获取MySQL版本号……183
8.6.2 获取用户名的函数……185
8.6.3 获取字符串的字符集和排序方式的函数……185

8.7 加密函数……186
8.8 窗口函数……187
8.9 MySQL函数的使用示例……188
8.10 本章小结……192
8.11 思考与练习……192

第 9 章 运算符……193
9.1 运算符概述……193
9.2 算术运算符……194
9.3 比较运算符……196
9.3.1 常用的比较运算符……197
9.3.2 特殊功能的比较运算符……199
9.4 逻辑运算符……203
9.5 位运算符……204
9.6 运算符的优先级……206
9.7 运算符综合示例……207
9.8 本章小结……209
9.9 思考与练习……210

第 10 章 视图……211
10.1 视图概述……211
10.1.1 视图的含义……211
10.1.2 视图的作用……212
10.2 创建视图……213
10.2.1 创建视图的语法形式……213
10.2.2 在单表上创建视图……214
10.2.3 在多表上创建视图……214
10.3 查看视图……215
10.3.1 使用DESCRIBE语句查看视图的基本信息……215
10.3.2 使用SHOW TABLE STATUS语句查看视图的基本信息……216
10.3.3 使用SHOW CREATE VIEW语句查看视图的详细信息……217
10.3.4 在views表中查看视图的详细信息……217
10.4 修改视图……218
10.4.1 使用CREATE OR REPLACE VIEW语句修改视图……218
10.4.2 使用ALTER语句修改视图……218
10.5 更新视图……219
10.6 删除视图……221
10.7 本章实战……221
10.8 本章小结……224
10.9 思考与练习……225

第 11 章 存储程序……226
11.1 创建、调用存储过程和函数……226
11.1.1 创建和调用存储过程……226
11.1.2 创建和调用存储函数……231
11.1.3 变量的使用……232
11.1.4 定义条件和处理程序……233
11.1.5 光标的使用……236
11.1.6 流程控制的使用……237
11.2 查看存储过程和函数……241
11.2.1 使用SHOW STATUS语句查看存储过程和函数的状态……241
11.2.2 使用SHOW CREATE语句查看存储过程和函数的定义……241
11.2.3 从information_schema.Routines表中查看存储过程和函数的信息……242
11.3 修改存储过程和函数……243
11.4 删除存储过程和函数……244
11.5 MySQL 8.0的全局变量的持久化……245
11.6 本章小结……246
11.7 思考与练习……246

第 12 章 触发器……247
12.1 创建触发器……247
12.1.1 创建只有一个执行语句的触发器……247
12.1.2 创建有多个执行语句的触发器……248
12.2 查看触发器……250
12.2.1 使用SHOW TRIGGERS语句查看触发器……250
12.2.2 在triggers表中查看触发器信息……252
12.3 触发器的使用……253
12.4 删除触发器……254
12.5 本章实战……254

12.6 本章小结……255
12.7 思考与练习……256

第 13 章 MySQL 权限与安全管理……257
13.1 权限表……257
13.1.1 user表……257
13.1.2 db表……259
13.1.3 tables_priv表和columns_priv表……261
13.1.4 procs_priv表……261
13.2 账户管理……262
13.2.1 登录和退出MySQL服务器……262
13.2.2 新建普通用户……263
13.2.3 删除普通用户……265
13.2.4 root用户修改自己的密码……266
13.2.5 root用户修改普通用户密码……266
13.3 权限管理……267
13.3.1 MySQL的各种权限……267
13.3.2 授权……269
13.3.3 收回权限……271
13.3.4 查看权限……272
13.4 访问控制……273
13.4.1 连接核实阶段……273
13.4.2 请求核实阶段……273
13.5 提升安全性的措施……274
13.5.1 AES 256加密……274
13.5.2 密码到期更换策略……276
13.5.3 安全模式安装……278
13.6 管理角色……278
13.7 本章实战……279
13.8 本章小结……282
13.9 思考与练习……282

第 14 章 数据备份与恢复……283
14.1 数据备份……283
14.1.1 使用MySQLdump命令备份……283
14.1.2 直接复制整个数据库目录……288
14.1.3 使用MySQLhotcopy工具快速备份……288
14.2 数据恢复……289
14.2.1 使用MySQL命令恢复……289
14.2.2 直接复制到数据库目录……289
14.2.3 MySQLhotcopy快速恢复……290
14.3 数据库迁移……290
14.3.1 相同版本的MySQL数据库之间的迁移……290
14.3.2 不同版本的MySQL数据库之间的迁移……291
14.3.3 不同数据库之间的迁移……291
14.4 表的导出和导入……291
14.4.1 使用SELECT...INTO OUTFILE导出文本文件……292
14.4.2 使用MySQLdump导出文本文件……295
14.4.3 使用MySQL导出文本文件……297
14.4.4 使用LOAD DATA INFILE方式导入文本文件……299
14.4.5 使用MySQLimport导入文本文件……300
14.5 本章实战……301
14.6 本章小结……305
14.7 思考与练习……305

第 15 章 MySQL 日志……306
15.1 日志简介……306
15.2 二进制日志……307
15.2.1 启动和设置二进制日志……307
15.2.2 查看二进制日志……308
15.2.3 删除二进制日志……309
15.2.4 使用二进制日志恢复数据库……310
15.2.5 暂时停止二进制日志功能……311
15.3 错误日志……311
15.3.1 启动和设置错误日志……311
15.3.2 查看错误日志……312
15.3.3 删除错误日志……312
15.4 通用查询日志……313
15.4.1 启动通用查询日志……313
15.4.2 查看通用查询日志……313
15.4.3 删除通用查询日志……314
15.5 慢查询日志……314

15.5.1 启动和设置慢查询日志……314
15.5.2 查看慢查询日志……315
15.5.3 删除慢查询日志……315
15.6 本章实战……315
15.7 本章小结……321
15.8 思考与练习……321

第 16 章 性能优化……322
16.1 优化简介……322
16.2 优化查询……323
16.2.1 分析查询语句……323
16.2.2 索引对查询速度的影响……325
16.2.3 使用索引查询……326
16.2.4 优化子查询……328
16.3 优化数据库结构……329
16.3.1 将字段很多的表分解成多个表……329
16.3.2 增加中间表……330
16.3.3 增加冗余字段……331
16.3.4 优化插入记录的速度……331
16.3.5 分析表、检查表和优化表……333
16.4 优化MySQL服务器……335
16.4.1 优化服务器硬件……335
16.4.2 优化MySQL的参数……335
16.5 临时表性能优化……336
16.6 服务器语句超时处理……338
16.7 创建全局通用表空间……338
16.8 本章实战……339
16.9 本章小结……341
16.10 思考与练习……341

第 17 章 综合项目……342
17.1 图书管理系统……342
17.1.1 需求管理……342
17.1.2 创建数据库……345
17.1.3 图书管理……349
17.1.4 用户信息管理……350
17.1.5 图书借阅管理……351
17.1.6 视图管理……352
17.2 网上购物系统……354
17.2.1 系统功能描述……354
17.2.2 系统功能分析……355
17.2.3 代码实现……357
17.2.4 程序运行……367
17.3 本章小结……369
17.4 思考与练习……369

参考文献……370

∞ 第 1 章 ∞

数据库基础

本章主要介绍数据库的相关概念，主要包括数据库技术的发展、数据库系统的组成、数据模型的概念与规范化、结构化查询语言(SQL)、数据库的体系结构、E-R 图的设计方法，以及数据库设计。通过本章的学习，读者可以对数据库基础知识有一个概括性的认识，并对数据库设计步骤有大致的了解。

本章的学习目标：

- 了解数据库技术的发展阶段、数据库系统的组成、数据模型与规范化、结构化查询语言(SQL)。
- 熟记数据库的体系结构，内容包括数据库三级模式结构、三级模式之间的映射。
- 掌握 E-R 图的概念及设计，内容包括实体、属性、关系的概念，以及 E-R 图设计原则。
- 掌握数据库设计的步骤与方法，内容包括为实体建立数据表、为表建立主键和外键、为字段选择合适的数据类型、定义约束条件。

1.1 数据库概述

1.1.1 数据库技术的发展

数据库技术应数据管理任务的需求而生。随着计算机技术的发展，数据管理技术也不断提高。数据管理技术的发展先后经历了人工管理、文件系统和数据库系统 3 个阶段。下面分别进行介绍。

1. 人工管理阶段

20 世纪 50 年代中期以前，计算机主要用于科学计算。当时的硬件设备和软件技术都很落后，数据管理主要通过手工进行。人工管理阶段具有如下特点。

- 数据量较少：数据和程序一一对应，数据主要服务于特定的应用程序，导致数据独立性很差。由于不同应用程序所处理的数据之间可能会有一定的关系，因此程序之间会有大量的重复数据。
- 数据不保存：因为该阶段计算机的主要任务是科学计算，一般不需要长期保存，计算出结果就行了。
- 没有软件系统对数据进行管理：程序员不仅要规定数据的逻辑结构，并且要在程序中设计物理结构，包括存储结构的存取方法、输入输出方式等。

2. 文件系统阶段

20 世纪 50 年代后期到 20 世纪 60 年代中期，硬件设备和软件技术都有了进一步发展，大容量

的磁盘等辅助存储设备的出现，使得专门管理辅助设备上的数据的文件系统应运而生，它是操作系统中的一个子系统。文件系统按照一定的规则将数据组织成为一个文件，应用程序通过文件系统对文件中的数据进行存取和加工。该阶段具有如下特点。

- 数据可以长期保留：程序可以按照文件名访问和读取数据，不必关心数据的物理位置。
- 数据不属于某个特定应用：应用程序和数据之间不再是直接的对应关系，数据可以重复使用。不同的应用程序无法共享同一数据文件。
- 文件组织形式的多样化：文件组织形式包括索引文件、链接文件、Hash 文件等。文件之间没有联系，相互独立，数据间的联系要通过程序去构造。
- 文件系统具有数据冗余、数据不一致性、数据孤立等缺点。

3. 数据库系统阶段

20 世纪 60 年代后期以来，计算机应用于管理系统，而且规模越来越大，应用越来越广泛，数据量急剧增长，对共享功能的要求越来越强烈，使用文件系统管理数据已经不能满足要求，于是出现了数据库系统来统一管理数据。

数据库系统是由计算机软件、硬件资源组成的系统，它实现了有组织地、动态地存储大量关联数据，并方便多用户访问。它与文件系统的重要区别是：数据的充分共享，交叉访问，与应用程序的高度独立性。这个阶段具有如下特点。

- 采用复杂的数据模型表示数据结构：数据模型描述数据本身的特点、数据之间的联系。数据不再面向单个应用，而是整个应用系统。数据冗余明显减少，实现数据共享。
- 有较高的数据独立性：数据库以一种更高级的组织形式，在应用程序和数据库之间由数据库管理系统(DBMS)负责访问数据来实现的。数据库对数据的存储是按照同一结构进行的，不同应用程序都可以直接操作这些数据。数据库对数据的完整性、唯一性、安全性都有一套有效的管理手段。

另外，数据库还提供管理和控制数据的各种简单操作命令，使用户编写程序更加容易。

1.1.2 数据库系统的组成

数据库系统(Database System，DBS)是采用数据库技术的计算机系统，是由数据库、数据库管理系统、数据库管理员(Database Administrator，DBA)、应用开发工具、应用系统、用户等组成，如图 1-1 所示。其中，数据库管理员是对数据库进行规划、设计、维护和监视等的专业管理人员，在数据库系统中起着非常重要的作用。

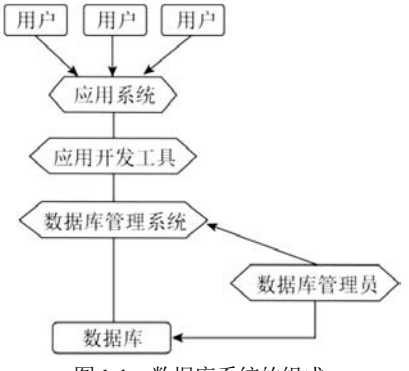

图 1-1 数据库系统的组成

1.1.3 数据模型与规范化

1. 数据模型的概念

数据模型(Data Model)是数据特征的抽象，它从抽象层次上描述了系统的静态特征、动态行为和约束条件，为数据库系统的信息表示与操作提供一个抽象的框架。数据模型所描述的内容有三部分，分别是数据结构、数据操作和完整性约束。

- 数据结构：主要描述数据的类型、内容、性质及数据间的联系等。数据结构是数据模型的基础，数据操作和约束都建立在数据结构上。不同的数据结构具有不同的操作和约束。
- 数据操作：主要描述在相应的数据结构上的操作类型和操作方式。
- 完整性约束：主要描述数据结构内数据间的语法、词义联系、它们之间的制约和依存关系，以及数据动态变化的规则，以保证数据的正确、有效和相容。

数据模型是数据库设计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。数据模型是数据库系统的核心和基础。

2. 不同应用层次的数据模型

数据模型按不同的应用层次分成 3 种类型，分别是概念数据模型、逻辑数据模型、物理数据模型。

1) 概念数据模型

概念数据模型(Conceptual Data Model)，是一种面向用户、面向客观世界的模型，主要用来描述现实世界的概念化结构。在数据库设计的初始阶段，数据库的设计人员应用该模型摆脱计算机系统及 DBMS 的具体技术问题的纠缠，集中精力分析数据及数据之间的联系等，与具体的数据库管理系统(Database Management System，DBMS)无关。概念数据模型必须转换成逻辑数据模型，才能在 DBMS 中实现。

在概念数据模型中最常用的是 E-R 模型、扩充的 E-R 模型、面向对象模型及谓词模型。其中 E-R 模型在构建数据库的过程中最常用，本章后续内容中将详细进行介绍。

2) 逻辑数据模型

逻辑数据模型(Logical Data Model)，是一种面向数据库系统的模型，是具体的 DBMS 所支持的数据模型，如层次数据模型、网状数据模型、关系数据模型。逻辑数据模型既要面向用户，又要面向系统，主要用于数据库管理系统(DBMS)的实现。

3) 物理数据模型

物理数据模型(Physical Data Model)，是一种面向计算机物理表示的模型，描述了数据在存储介质上的组织结构，它不但与具体的 DBMS 有关，而且还与操作系统和硬件有关。每一种逻辑数据模型在实现时都有其对应的物理数据模型。DBMS 为了保证其独立性与可移植性，大部分物理数据模型的实现工作由系统自动完成，而设计者只设计索引、聚集等特殊结构。

3. 常见的逻辑数据模型

前面提到过 3 种基本的逻辑数据模型，它们是层次模型、网状模型和关系模型。这三种逻辑数据模型是按其数据结构而命名的。前两种采用格式化的结构。在这类结构中实体用记录型表示，而记录型抽象为图的顶点。记录型之间的联系抽象为顶点间的连接弧。整个数据结构与图相对应。其中，层次模型的基本结构是树形结构；网状模型的基本结构是一个不加任何限制条件的无向图。关系模型为非格式化的结构，用单一的二维表的结构表示实体及实体之间的联系，关系模型是目前数据库中常用的数据模型。

1) 层次模型

层次模型将数据组织成一对多关系的结构，用树形结构表示实体及实体间的联系，如图 1-2 所示。在层次模型中，每棵树有且仅有一个无双亲节点，称为根；树中除根外所有节点有且仅有一个双亲。

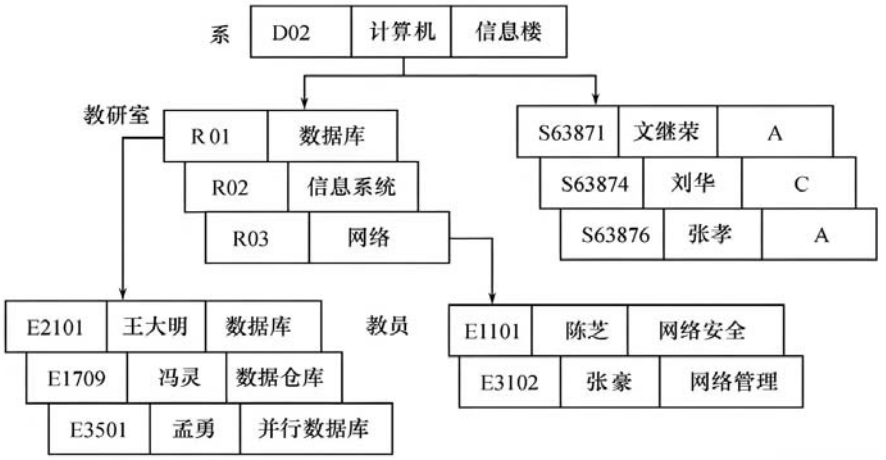

图 1-2　层次模型

2) 网状模型

用有向图结构表示实体类型及实体间联系的数据模型被称为网状模型，如图 1-3 所示。用网状模型编写应用程序极其复杂，数据的独立性较差。

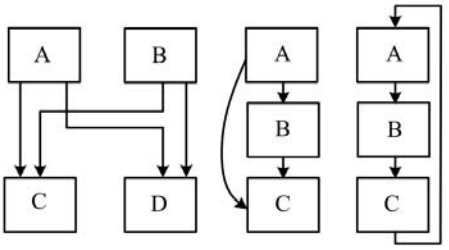

图 1-3　网状模型

3) 关系模型

关系模型以二维表来描述数据。在关系模型中，每个表都有多个字段列和记录行，而每个字段列都有固定的属性(如数字、字符、日期等)，如图 1-4 所示。关系模型数据结构简单、清晰，具有很高的数据独立性，是目前主流的数据库数据模型。关系模型的基本术语如下。

- 关系：一个二维表就是一个关系。
- 元组：二维表中的一行，即表中的记录。
- 属性：二维表中的一列，用类型和值表示。
- 域：每个属性取值的变化范围，如性别的域为{男，女}。

关系中的数据约束如下。

- 实体完整性约束：约束关系的主键中属性值不能为空值。
- 参照完整性约束：关系之间的基本约束。
- 用户定义的完整性约束：反映了具体应用中数据的语义要求。

学生信息表

学生姓名	年级	家庭住址
张三	2023	朝阳区朝阳北路
李四	2023	东城区建国门大街
王五	2023	海淀区中关村东路

成绩表

学生姓名	课程	成绩
张三	数学	99
张三	语文	95
张三	英语	98
李四	数学	100
李四	语文	92
李四	英语	93
王五	数学	100
王五	语文	95
王五	英语	99

图 1-4　关系模型

4. 关系数据库的规范化

关系数据库的规范化理论是，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，可以分为 5 个等级：第一范式(1NF)，第二范式(2NF)，……，第五范式(5NF)。其中，NF 是 normal form 的缩写。一般情况下，只要把数据规范到第三范式标准就可以满足需要。下面举例介绍前 3 种范式。

1) 第一范式

第一范式是指在一个关系中，消除重复字段，且各字段都是最小的逻辑存储单位。第一范式是第二范式和第三范式的基础，是最基本的范式。第一范式包括下列指导原则。

- 关系中的每个元组的每个属性只可以包含一个值。
- 关系中的每个元组必须包含相同数量的值。
- 关系中的每个元组一定不能相同。

在任何一个关系数据库中，第一范式是对关系模式的基本要求，不满足第一范式的数据库就不是关系数据库。

如果数据表中的每一列都是不可再分割的基本数据项，即同一列中不能有多个值，那么就称此数据表符合第一范式，由此可见第一范式具有不可再分解的原子特性。

在第一范式中，数据表的每一行只包含一个实体的信息，并且每一行的每一列只能存放实体的一个属性。例如，对于学生信息，不可以将学生实体的两个或多个或所有属性信息(如学号、姓名、性别、年龄、班级等)都放在一个列中予以显示，也就是说，一个列中应只存放学生实体的一个属性信息。

如果数据表中的列信息都符合第一范式，那么在数据表中的字段都是单一的、不可再分的。如表 1-1 就是不符合第一范式的学生信息表，因为“班级”列中包含“系别”和“班级”两个属性信息，这样“班级”列中的信息就不是单一的，是可以再分的；而表 1-2 就是符合第一范式的学生信息表，它将原“班级”列的信息拆分到“系别”列和“班级”列中。

表 1-1　不符合第一范式的学生信息表

学号	姓名	性别	年龄	班级
9527	东*方	男	20	计算机系 3 班

表 1-2　符合第一范式的学生信息表

学号	姓名	性别	年龄	系别	班级
9527	东*方	男	20	计算机	3 班

2) 第二范式

第二范式是在第一范式的基础上建立起来的，即满足第二范式必先满足第一范式。第二范式要求数据库表中的每个实体(即各个记录行)必须可以被唯一地区分。为实现区分各行记录，通常需要为表设置一个“区分列”，用以存储各个实体的唯一标识。在学生信息表中，设置了“学号”列，由于每个学生的编号都是唯一的，因此每个学生可以被唯一地区分(即使学生存在重名的情况)，这个唯一属性列被称为主关键字或主键。

第二范式要求实体的属性完全依赖于主关键字，即不能存在仅依赖于主关键字一部分的属性，如果存在，那么这个属性和主关键字的这一部分应该被分离出来形成一个新的实体，新实体与原实体之间是一对多的关系。

例如，以员工工资信息表为例，若以员工编码、岗位为组合关键字(即复合主键)，就会存在如下决定关系。

(员工编码、岗位)→(决定)(姓名、年龄、学历、基本工资、绩效工资、奖金)

在上面的决定关系中，还可以进一步被拆分为如下两种决定关系。

(员工编码)→(决定)(姓名、年龄、学历)
(岗位)→(决定)(基本工资)

其中，员工编码决定了员工的基本信息(包括姓名、年龄、学历等)，而岗位决定了基本工资，因此这个关系表不满足第二范式。

对于上面的这种关系，可以把上述两个关系表更改为如下 3 个表。

- 员工信息表：EMPLOYEE(员工编码、姓名、年龄和学历)。
- 岗位工资表：QUARTERS(岗位和基本工资)。
- 员工工资表：PAY(员工编码、岗位、绩效工资和奖金)。

3) 第三范式

第三范式是在第二范式的基础上建立起来的，即满足第三范式必先满足第二范式。第三范式要求关系表不存在非关键字列对任意候选关键字列的传递函数依赖，也就是说，第三范式要求一个关系表中不包含已在其他表中包含的非主关键字信息。

所谓传递函数依赖，是指如果存在关键字段 A 决定非关键字段 B，而非关键字段 B 决定非关键字段 C，则称非关键字段 C 传递函数依赖于关键字段 A。

例如，这里以员工信息表(EMPLOYEE)为例，该表中包含员工编码、员工姓名、年龄、部门编码、部门经理等信息，该关系表的关键字为“员工编码”，因此存在如下决定关系。

(员工编码)→(决定)(员工姓名、年龄、部门编码、部门经理)

上面的这个关系表是符合第二范式的，但它不符合第三范式，因为该关系表内部隐含着如下决

定关系。

(员工编码)→(决定)(部门编码)→(决定)(部门经理)

上面的关系表存在非关键字段“部门经理”对关键字段“员工编码”的传递函数依赖。对于上面的这种关系，可以把这个关系表(EMPLOYEE)更改为如下两个关系表。

- 员工信息表：EMPLOYEE(员工编码、员工姓名、年龄和部门编码)。
- 部门信息表：DEPARTMENT(部门编码和部门经理)。

对于关系数据库的设计，理想的设计目标是按照“规范化”原则存储数据，因为这样做能够消除数据冗余、更新异常、插入异常和删除异常。

5. 关系数据库的设计原则

数据库设计是指对于一个给定的应用环境，根据用户的需求，利用数据模型和应用程序模拟现实世界中该应用环境的数据结构和处理活动的过程。

数据库设计原则如下。

- 数据库内数据文件的数据组织应获得最大限度的共享、最小的冗余度，消除数据及数据依赖关系中的冗余部分，使依赖于同一个数据模型的数据达到有效的分离。
- 保证输入、修改数据时数据的一致性与正确性。
- 保证数据与使用数据的应用程序之间的高度独立性。

1.1.4　结构化查询语言(SQL)

结构化查询语言(Structured Query Language，SQL)是一种应用于关系数据库查询的结构化语言，最早是由博伊斯(Boyce)和钱柏林(Chamberlin)在 1974 年提出的，称为 SEQUEL 语言。

1976 年，IBM 公司的 San Jose 研究所在研制关系数据库管理系统 System R 时将其修改为 SEQUEL 2，即目前的 SQL 语言。同年，SQL 开始在商品化关系数据库管理系统中得到应用。

1982 年，美国国家标准化组织(ANSI)确认 SQL 为数据库系统的工业标准。SQL 是一种介于关系代数和关系演算之间的语言，具有丰富的查询功能，同时具有数据定义和数据控制功能，是集数据定义、数据查询和数据控制于一体的关系数据语言。目前，有许多关系数据库管理系统支持 SQL 语言，如 SQL Server、Access、Oracle、MySQL、DB2 等。

SQL 语言的功能包括数据查询、数据操纵、数据定义和数据控制 4 部分。SQL 语言简洁、方便、实用，完成其核心功能只用了 6 个动词——SELECT、CREATE、INSERT、UPDATE、DELETE 和 GRANT(REVOKE)。作为关系数据库的标准语言，它已被众多商用数据库管理系统产品所采用，成为应用最广的关系数据库语言。不过，不同的数据库管理系统在其实践过程中都对 SQL 规范做了某些编改和扩充。所以，实际上不同数据库管理系统之间的 SQL 不能完全相互通用。例如，甲骨文公司的 Oracle 数据库所使用的 SQL 是 Procedural Language /SQL(简称 PL/SQL)，而微软公司的 SQL Server 数据库系统支持的是 Transact-SQL(简称 T-SQL)。MySQL 也对 SQL 标准进行了扩展，只是至今没有命名。

1.2　数据库的体系结构

数据库的体系结构包括三级模式(外模式、模式、内模式)和两级映射(外模式/概念模式映射、概

念模式/内模式映射)。如图 1-5 所示。

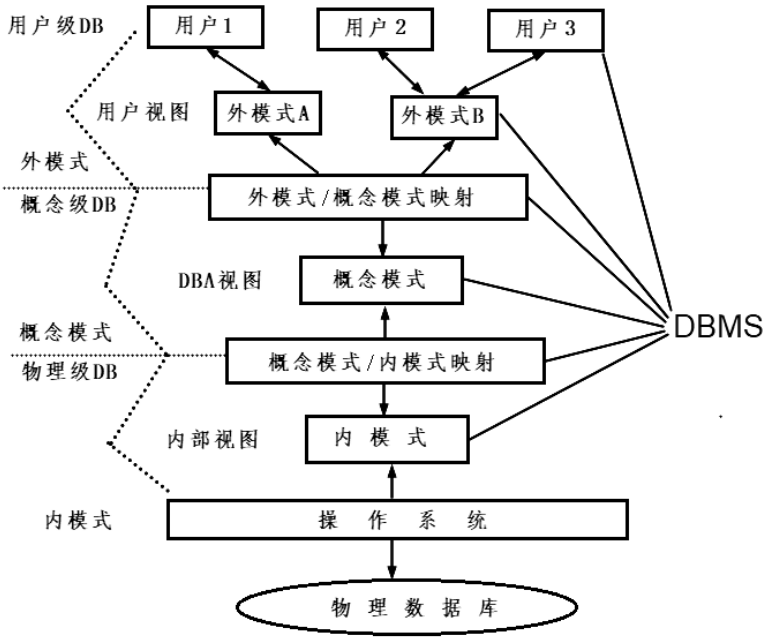

图 1-5　数据库三级模式结构

1.2.1　数据库三级模式

数据库的三级模式结构是指外模式、模式和内模式。

1. 外模式

外模式也被称为用户模式。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，也是数据库用户的数据视图。此外，它还是与某一应用有关的数据的逻辑表示。外模式是模式的子集，一个数据库可以有多个外模式。定义外模式是保证数据安全性的一个有力措施。

2. 模式

模式也被称为逻辑模式或概念模式。它是数据库中全体数据的逻辑结构和特征的描述，也是所有用户的公共数据视图。一个数据库只有一个模式。模式处于三级结构的中间层。注意定义模式时不仅要定义数据的逻辑结构，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。

3. 内模式

内模式也被称为存储模式。它是数据物理结构和存储方式的描述，也是数据在数据库内部的表示方式。一个数据库只有一个内模式。

1.2.2　三级模式之间的映射

为了能够在内部实现数据库的 3 个抽象层次的联系和转换，数据库管理系统在三级模式之间提

供了两层映射，分别为外模式/概念模式映射和概念模式/内模式映射。

1. 外模式/概念模式映射

同一个概念模式可以有任意多个外模式。对于每一个外模式，数据库系统都有一个外模式/概念模式映射。当概念模式发生改变时，由数据库管理员对各个外模式 / 概念模式映射做相应的改变，可以使外模式保持不变。这样，依据数据外模式编写的应用程序就不用修改，保证了数据与程序的逻辑独立性。

2. 概念模式/内模式映射

数据库中只有一个概念模式和一个内模式，因此概念模式/内模式映射是唯一的，它定义了数据库的全局逻辑结构与存储结构之间的对应关系。当数据库的存储结构发生变化时，数据库管理员通过调整概念模式与内模式之间的映射关系，确保应用程序看到的仍然是稳定的数据逻辑结构。从而实现了数据与程序的物理独立性。

1.3 E-R 图

E-R 图(Entity-Relationship Diagram)也称“实体—关系图”，用于描述现实世界的事物，以及事物与事物之间的关系。其中 E 表示实体，R 表示关系。它提供了表示实体类型、属性和关系的方法。下面将详细介绍实体、属性、关系，以及 E-R 图的设计原则。

1.3.1 实体和属性

在数据库领域中，客观世界中的万事万物都被称为实体。实体既可以是指客观存在并可相互区别的事物，例如高山、流水、学生、老师等，又可以是一些抽象的概念或地理名词，例如精神生活、物质基础、吉林省、北京市等。实体的特征(外在表现)称为属性，通过属性可以区分同类实体。例如，一本书可以具备下列属性：书名、大小、封面颜色、页数、出版社等，并且根据这些属性可以在一堆图书中找到所要的图书。

在通常情况下，开发人员在设计 E-R 图时，使用矩形表示实体，在矩形框内写实体名(实体名是每个实体的唯一标识)，使用椭圆表示属性，并且使用无向边将其与实体连接起来。

【例 1-1】设计图书馆管理系统的图书实体图。在图书馆管理系统中，图书是一个实体，它包括编号、条形码、书名、类型、作者、译者、出版社、价格、页码、书架、录入时间、操作员和是否删除等属性。对应的实体图如图 1-6 所示。

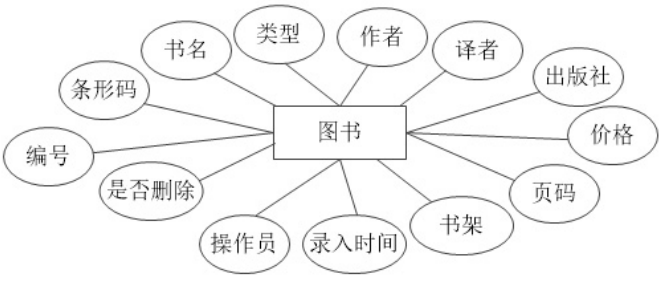

图 1-6　图书馆管理系统的图书实体图

说明：

在图书馆的图书实体的属性中，“是否删除”属性用于标记图书是否被删除，由于图书馆中的图书信息不可以被随意删除，因此即使当某种图书不能再借阅，而需要删除其档案信息时，也只能采用设置删除标记的方法。

1.3.2 关系

在客观世界中，实体并不是孤立存在的，实体与实体之间通常还存在一些联系。在 E-R 图中，可以使用关系表示实体间的联系。通常使用菱形表示实体间的关系，在菱形框内写明联系名，并且使用无向边将其与有关的实体连接起来。同时，还需要在无向边旁标上关系的类型。

在通常情况下，实体间存在以下 3 种关系。

1. 一对一关系

一对一关系是指两个实体 A 和 B，如果 A 中的每一个值在 B 中至多有一个实体值与其对应，反之亦然，那么则称 A 和 B 为一对一关系。在 E-R 图中，使用 1∶1 表示。例如，在一个图书馆中，只能有一个馆长，反之，一个馆长只能在一个图书馆任职。

2. 一对多关系

一对多关系是指两个实体 A 和 B，如果 A 中的每一个值在 B 中有多个实体值与其对应，反之在 B 中每一个实体值在 A 中至多有一个实体值与之对应，那么则称 A 和 B 为一对多关系。在 E-R 图中，使用 1∶*n* 表示。例如，在图书馆中，一个书架上可以放置多本图书，但是一本图书只能放置在一个书架上。因此，书架和图书之间存在一对多的关系。

3. 多对多关系

多对多关系是指两个实体 A 和 B，如果 A 中的每一个值在 B 中有多个实体值与其对应，反之亦然，那么则称 A 和 B 为多对多关系。在 E-R 图中，使用 *m*∶*n* 表示。例如，在图书馆中，一个读者可以借阅多本图书，反之，一本图书也可以被多个读者借阅。因此，读者和图书之间存在多对多的关系。

1.3.3 E-R 图设计原则

E-R 图的设计虽然没有一个绝对固定的方法，但一般情况下，需要遵循以下基本原则。

- 先设计局部 E-R 图，再把每一个局部的 E-R 图综合起来，生成总体的 E-R 图。
- 属性应该存在于且只存在于某一个实体或者关系中，这样可以避免数据冗余。例如，在图 1-7 所示的 E-R 图中，就出现了大量的数据冗余，所借图书属性不能重复。

图 1-7 存在冗余的读者实体图

- 实体是一个单独的个体，不能存在于另一个实体中，即不能作为另一个实体的属性。例如，图 1-6 所示的图书实体，不能作为借阅实体的一个属性。同一个实体在同一个 E-R 图中只能出现一次。

【例 1-2】设计图书馆管理系统的 E-R 图。在图书馆管理系统中，主要包括两个实体和两个关系，两个实体分别是图书和读者；两个关系分别是借阅和归还，这两个关系都是多对多的关系。对应的 E-R 图如图 1-8 所示。

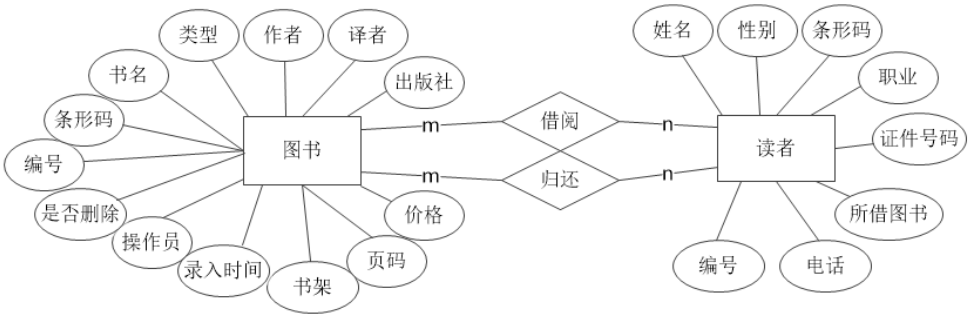

图 1-8　图书馆管理系统的 E-R 图

1.4 数据库设计

在设计出 E-R 图后，就可以根据该 E-R 图生成对应的数据表，具体步骤如下。

(1) 为 E-R 图中的每一个实体创建一张对应的数据表。

(2) 为每张数据表定义主键(一般情况下，会将作为唯一标识的编号作为主键)或者外键。

(3) 除了为实体创建对应的数据表，还要为多对多关系建立数据表，如图书和读者之间形成多对多(m : n)的借阅和归还关系，则需要为借阅和归还分别建立数据表。

(4) 为字段选择合适的数据类型。

(5) 定义约束条件(可选)。

下面将进行详细介绍。

1.4.1 为实体建立数据表

在 E-R 图中，每个实体通常对应一张数据表，实体的属性对应于数据表中的字段。在程序的开发过程中，考虑到程序的兼容性，通常使用英文的字段名，所以在转换的过程中，经常需要将中文的属性名转换为对应意义的英文。例如，可以将“书名”属性转换为 bookname。

【例 1-3】根据图 1-8 所示的图书实体，可以得到包含编号、条形码、书名、类型、作者、译者、出版社、价格、页码、书架、录入时间、操作员和是否删除 13 个字段的图书信息表，对应的结构如下：

```
tb_book(id,barcode,bookname,typeid,author,translator,ISBN,page,rice,bookcase,inTime,operator,del)
```

1.4.2 为表建立主键和外键

由于在设计数据表时，不允许出现完全相同的两条记录，因此通常会创建一个关键字(Key)字段，

用于唯一标识数据表中的每一条记录。例如，在读者信息表中，由于条形码不允许重复且不允许为空，因此条形码可以作为读者信息表中的关键字。另外，在读者信息表中，还存在一个编号字段，该字段也不允许重复且不允许为空，所以编号字段也可以作为读者信息表中的关键字。

1. 建立主键

在设计数据库时，为每个实体建立对应的数据表后，通常还会为其创建主键，主键所在的表称为主表。一般情况下，主键都是在所有的关键字中选择的。在选择主键时，一般遵循以下两条原则。

- 作为主键的关键字可以是一个字段，也可以是字段的组合。
- 作为主键的字段的值必须具有唯一性，并且不能为空(null)。如果主键由多个字段构成时，那么这些字段都不能为空。

例如，在创建图书信息表时，由于编号字段不能重复，并且不能为空，因此 id 字段可以作为主键。添加主键后，tb_book 的结构如下(加下画线的字段为主键)：

```
tb_book(id,barcode,bookname,typeid,author,translator,ISBN,page,rice,bookcase,inTime,operator,del)
```

2. 建立外键

如果存在两张数据表，表 T1 中的一个字段 fk 对应于表 T2 的主键 pk，那么字段 fk 称为表 T1 的外键，T1 称为外键表或子表。此时，表 T1 的字段 fk，要么是表 T2 的主键 pk 的值，要么是空值。外键通常用于实现参照完整性。例如，存在一对多关系的两个实体“图书”和“出版社”，它们转换为数据表后，对应的主外键关系如下：

```
tb_book(id,barcode,bookname,typeid,author,translator,ISBN,page,rice,bookcase,inTime,operator,del)
tb_publisher(ISBN, pubname)
```

其中，在 tb_book 表中，ISBN 为外键；在 tb_publisher 表中，ISBN 为主键。

1.4.3 为多对多关系建立数据表

除了为实体创建对应的数据表，还要为多对多关系建立数据表，如图书和读者之间形成多对多($m:n$)的借阅和归还关系，则需要为借阅和归还分别建立数据表，具体如下：

```
tb_borrow(borrow_id,reader_id,book_id,borrowTime,backTime,operator,ifback)
tb_back(back_id,reader_id,book_id,backTime,operator)
```

其中，tb_borrow 表的主键为 borrow_id，tb_back 表的主键为 back_id。两个表中 reader_id、book_id 为外键。

1.4.4 为字段选择合适的数据类型

在数据库设计过程中，为字段选择合适的数据类型也非常重要。合适的数据类型可以有效地节省数据库的存储空间、提升数据的计算性能、节省数据的检索时间。在数据库管理系统中，常用的数据类型包括字符串类型、数值类型和日期时间类型。下面分别进行介绍。

1. 字符串类型

字符串类型用于保存一系列的字符。这些字符在使用时是采用单引号括起来的，主要用于保存不参与运算的信息。例如，图书名称'HTML 5 从入门到精通'、条形码'9787302210337'和 ISBN'302'

都属于字符串类型。虽然后面两个在外观上看是整数，但是这些整数只是显示用的，不参与计算，所以也设置为字符串类型。字符串类型可以分为定长字符串类型和变长字符串类型。其中，定长字符串类型保存的数据长度都一样，如果输入的数据没有达到要求的长度，那么会自动用空格补全；而变长字符串类型保存的数据长度与输入的数据相同(前提是输入的数据不超出该字段设置的长度)。

2. 数值类型

数值类型是指可以参与算术运算的类型。它可以分为整型和小数类型，其中小数类型又包括浮点型和双精度型。例如，图书的本数就可以设置为整型，而图书的单价就需要设置为浮点型。

3. 日期时间类型

日期时间类型是指用于保存日期或者时间的数据类型，通常可以分为日期类型、时间类型和日期时间类型。其中，日期类型存储的数据是“YYYY-MM-DD”格式的字符串；时间类型存储的数据是“hh:mm:ss”格式的字符串；日期时间类型存储的数据是“YYYY-MM-DD hh:mm:ss”格式的字符串。例如，图书借阅时间就可以设置为日期时间类型，因为需要存储日期和时间。

1.4.5　定义约束条件

在设计数据库时，还可能需要为数据表设置一些约束条件，从而保证数据的完整性。常用的约束条件有以下 6 种。

- 主键约束：用于约束唯一性和非空性，通过为表设置主键实现。一张数据表中只能有一个主键。在数据录入过程中主键字段必须唯一，并且不能为空。
- 外键约束：需要建立两张数据表间的关系，并且引用主表的字段。外键字段的数据要么是主键字段的某个值，要么为空。在建立关系时，主表和子表通过外键关联。
- 唯一性约束：用于约束唯一性，可以通过为表设置唯一性约束实现。满足唯一性约束的字段可以为空。
- 非空约束：用于约束表中的某个字段不能为空。
- 检查约束：用于检查字段的输入值是否满足指定的条件。如果输入的数据与指定的字段类型不匹配，那么该数据将不能被写入数据库中。对于这个约束，一般的数据库管理系统都会自动检查。
- 默认值约束：用于为字段设置默认值。当输入数据时，如果该字段没有输入任何内容，那么会自动填入指定的默认值。

1.5　本章小结

本章主要介绍数据库技术基础知识，首先概括性介绍了数据库，包括数据库技术的发展、数据库系统的组成、数据模型与规范化、结构化查询语言(SQL)；然后介绍了数据库体系结构，包括数据库三级模式(外模式、模式、内模式)和两级映射(外模式/概念模式映射和概念模式/内模式映射)；接着介绍了 E-R 图的绘制；最后介绍了数据库设计过程。其中，E-R 图和数据库设计是本章的实践重点，希望大家认真学习，重点掌握。

1.6 思考与练习

1. 数据库技术的发展经历了哪些阶段？
2. 简述数据库与数据库管理系统的概念。
3. 简单描述结构化查询语言(SQL)。
4. 简单描述数据模型的组成。
5. 常用的数据库数据模型主要有哪几种？
6. 简单描述 E-R 图及其构成元素。
7. 简述 E-R 图的设计原则。
8. 简单描述根据 E-R 图生成对应的数据表的基本步骤。