



高等学校计算机教材

Oracle

实用教程 (第4版) (Oracle 12c版)

◎ 郑阿奇 主编
◎ 周 敏 张 洁 编著



中国工信出版集团



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

高等学校计算机教材

Oracle 实用教程

(第4版) (Oracle 12c 版)

郑阿奇 主编
周 敏 张 洁 编著

电子工业出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

本书以 Oracle 12c (中文版) 作为平台, 分别介绍 Oracle 数据库管理系统和在当前流行平台上开发 Oracle 数据库应用系统。Oracle 数据库管理系统介绍 Oracle 12c 的主要功能, 语法为中文层次格式, 实例数据库表字段名为汉字, 方便教学。流行平台包括 PHP 5、Java EE 7 和 Visual C# 2013。综合应用实习数据准备通过创建实习用数据库及其对象, 简单小结 Oracle 数据库的基本对象创建和操作, 不同平台操作同样的数据库, 实现同样功能, 给读者带来了极大的方便。

本书可作为大学本科、高职高专有关课程教材, 也可供广大 Oracle 数据库应用开发人员使用或参考。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有, 侵权必究。

图书在版编目 (CIP) 数据

Oracle 实用教程: Oracle 12c 版 / 郑阿奇主编; 周敏, 张洁编著. —4 版. —北京: 电子工业出版社, 2015.11
高等学校计算机教材
ISBN 978-7-121-27380-3

I. ①O… II. ①郑… ②周… ③张… III. ①关系数据库系统—高等学校—教材 IV. ①TP311.138

中国版本图书馆 CIP 数据核字 (2015) 第 241289 号

策划编辑: 程超群

责任编辑: 郝黎明 特约编辑: 张燕虹

印 刷: 三河市鑫金马印装有限公司

装 订: 三河市鑫金马印装有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1092 1/16 印张: 21.75 字数: 626 千字

版 次: 2003 年 10 月第 1 版

2015 年 11 月第 4 版

印 次: 2015 年 11 月第 1 次印刷

定 价: 49.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前 言

Oracle 是目前世界上使用最为广泛的关系型数据库管理系统，在数据库领域一直处于领先地位，在信息系统管理、企业数据处理、电子商务、网络、大数据和云计算等各个方面广泛应用，最新版本为 Oracle 12c。

本书以 Oracle 12c（中文版）为平台，分别介绍 Oracle 数据库管理系统和在当前流行平台上开发 Oracle 数据库应用系统。在介绍数据库基础后，系统介绍 Oracle 12c 的主要功能。包括数据库和表的创建和操作，数据库的查询和视图，索引与数据完整性，PL/SQL 语言，存储过程和触发器，系统安全管理，表空间、备份和恢复，事务、锁、闪回和 Undo 表空间以及 Oracle 12c 的其他概念。流行开发平台为 3 种，开发的 Oracle 数据库应用系统均为学生成绩管理系统。

本书主要特点如下。

（1）介绍数据库最基本的原理、驱动、接口，从总体上理清思路，便于学生理解。

（2）Oracle 基础部分采用中文分层命令格式，从而使语法描述变得简洁，更容易掌握。实例数据库表字段名为汉字，方便教学。

（3）运行结果屏幕化，一般不会出现命令错误。内容层次更加清楚，由浅及深，易于掌握。

（4）综合应用实习数据准备通过创建实习用数据库及其对象，简单小结 Oracle 数据库的基本对象创建和操作。

（5）综合应用包含目前最流行的 PHP 5、Java EE 7 和 Visual C# 2013 等平台操作 Oracle 12c 数据库。每个平台都介绍了连接数据库的基本知识、操作 Oracle 数据库的主要方法，并且构成了一个小的应用系统。网上提供源代码和数据库免费下载，很容易让读者模仿和掌握开发 Oracle 数据库应用系统。

本书不仅适合教学，也适合 Oracle 的各类培训和用 Oracle 编程开发应用程序的用户学习和参考。

本书由南京师范大学郑阿奇主编，周敏和张洁编著。参加本书编写的还有梁敬东、丁有和、曹弋、陈瀚、徐文胜、殷红先、张为民、姜乃松、钱晓军、彭作民、高茜、陈冬霞、徐斌、王志瑞、孙德荣、周怡明、刘博宇、郑进、周何骏、陶卫冬、严大牛、邓拼搏、俞琰、吴明祥、于金彬、马骏等。此外，还有许多同志对本书提供了很多帮助，在此一并表示感谢！

由于作者水平有限，不当之处在所难免，恳请读者批评指正。

意见建议邮箱：easybooks@163.com

作 者

2015 年 8 月

目 录

第 1 部分 实用教程

第 1 章 数据库的基本概念	1	3.3.3 删除表	43
1.1 数据库	1	3.3.4 插入记录	44
1.1.1 数据库、数据库管理系统和 数据库系统	1	3.3.5 删除记录	47
1.1.2 数据模型	2	3.3.6 修改记录	48
1.1.3 关系型数据库语言	3	第 4 章 数据库的查询和视图	50
1.2 数据库设计	4	4.1 选择、投影和连接	50
1.2.1 概念结构设计	4	4.1.1 选择	50
1.2.2 逻辑结构设计	6	4.1.2 投影	50
1.2.3 物理结构设计	7	4.1.3 连接	51
1.3 数据库应用系统	7	4.2 数据库的查询	52
1.3.1 C/S 模式应用系统	7	4.2.1 选择列	52
1.3.2 B/S 模式应用系统	8	4.2.2 选择行	56
第 2 章 Oracle 12c 数据库介绍	9	4.2.3 查询对象	63
2.1 Oracle 简介与安装	9	4.2.4 连接	64
2.1.1 Oracle 数据库简介	9	4.2.5 汇总	67
2.1.2 Oracle 12c 的安装	9	4.2.6 排序	71
2.2 Oracle 数据库的基本结构	14	4.2.7 合并	72
2.2.1 内部结构	14	4.3 数据库视图	72
2.2.2 外部结构	17	4.3.1 视图的概念	72
2.2.3 数据库实例	18	4.3.2 创建视图	73
2.3 Oracle 12c 数据库工具	21	4.3.3 查询视图	75
第 3 章 数据库的创建和操作	25	4.3.4 更新视图	76
3.1 以界面方式创建数据库 (采用 DBCA)	25	4.3.5 修改视图的定义	77
3.2 以界面方式操作数据库 (采用 SQL Developer)	30	4.3.6 删除视图	78
3.2.1 表操作	30	4.4 含替换变量的查询	79
3.2.2 表数据操作	35	4.4.1 &替换变量	79
3.2.3 执行 SQL 命令	38	4.4.2 &&替换变量	81
3.3 命令方式操作数据库 (采用 SQL*Plus)	39	4.4.3 DEFINE 和 ACCEPT 命令	81
3.3.1 创建表	39	第 5 章 索引与数据完整性	83
3.3.2 修改表	41	5.1 索引	83
		5.1.1 索引的分类	83
		5.1.2 使用索引的原则	84
		5.1.3 创建索引	85
		5.1.4 维护索引	87

5.1.5 删除索引	88	6.8.4 Oracle 12c 内置包	141
5.2 数据完整性	89	6.9 集合	141
5.2.1 数据完整性的分类	89	6.9.1 联合数组	141
5.2.2 域完整性的实现	91	6.9.2 嵌套表	143
5.2.3 实体完整性的实现	94	6.9.3 可变数组	144
5.2.4 参照完整性的实现	97	6.9.4 集合的属性和方法	145
第 6 章 PL/SQL 语言	101	第 7 章 存储过程和触发器	150
6.1 PL/SQL 概述	101	7.1 存储过程	150
6.1.1 PL/SQL 的组成	101	7.1.1 存储过程的创建	150
6.1.2 PL/SQL 的特点	102	7.1.2 存储过程的调用	152
6.1.3 PL/SQL 的开发和运行环境	104	7.1.3 存储过程的修改	154
6.2 PL/SQL 字符集	104	7.1.4 存储过程的删除	154
6.2.1 合法字符	104	7.2 触发器	155
6.2.2 运算符	104	7.2.1 以命令方式创建触发器	155
6.2.3 其他符号	106	7.2.2 以界面方式创建触发器	159
6.3 PL/SQL 变量、常量和数据类型	107	7.2.3 启用和禁用触发器	160
6.3.1 变量	107	7.2.4 触发器的删除	161
6.3.2 常量	109	第 8 章 系统安全管理	162
6.3.3 常用数据类型	109	8.1 用户	162
6.3.4 对象类型	110	8.1.1 创建用户	162
6.3.5 数据类型转换	111	8.1.2 管理用户	167
6.4 PL/SQL 基本程序结构和语句	111	8.2 权限管理	168
6.4.1 PL/SQL 程序块	111	8.2.1 权限概述	168
6.4.2 条件结构	112	8.2.2 系统权限管理	169
6.4.3 循环结构	115	8.2.3 对象权限管理	173
6.4.4 选择和跳转语句	118	8.2.4 安全特性	174
6.4.5 异常	120	8.3 角色管理	176
6.5 系统内置函数	123	8.3.1 角色概述	176
6.6 用户定义函数	127	8.3.2 创建用户角色	177
6.6.1 创建函数	127	8.3.3 管理用户角色	178
6.6.2 调用函数	129	8.4 概要文件和数据字典视图	179
6.6.3 删除函数	129	8.4.1 创建概要文件	179
6.7 游标	129	8.4.2 修改概要文件	181
6.7.1 显式游标	130	8.4.3 数据字典视图	181
6.7.2 隐式游标	132	8.5 审计	183
6.7.3 游标 FOR 循环	133	8.5.1 登录审计	183
6.7.4 游标变量	134	8.5.2 操作审计	184
6.8 包	136	8.5.3 对象审计	185
6.8.1 包的创建	136	8.5.4 权限审计	186
6.8.2 包的初始化	139	第 9 章 表空间、备份和恢复	187
6.8.3 重载	140	9.1 表空间	187

9.1.1 创建表空间	187	10.3.1 基本概念	215
9.1.2 管理表空间	189	10.3.2 查询闪回	215
9.1.3 删除表空间	191	10.3.3 表闪回	216
9.2 备份和恢复概述	191	10.3.4 删除闪回	218
9.2.1 备份概述	191	10.3.5 数据库闪回	220
9.2.2 恢复概述	194	10.3.6 归档闪回	224
9.3 导入/导出	195	10.4 Undo 表空间	225
9.3.1 导出	195	10.4.1 自动 Undo 管理	225
9.3.2 导入	197	10.4.2 Undo 表空间的优点	226
9.4 脱机备份	200	10.4.3 Undo 表空间管理参数	226
9.5 联机备份	201	10.4.4 创建和管理 Undo 表空间	226
9.5.1 以 ARCHIVELOG 方式运行 数据库	201	第 11 章 其他概念	229
9.5.2 执行数据库联机备份	202	11.1 同义词	229
9.6 数据泵	202	11.1.1 创建同义词	229
9.6.1 概述	202	11.1.2 使用同义词	230
9.6.2 EXPDP 导出	202	11.1.3 删除同义词	230
9.6.3 IMPDP 导入	204	11.2 数据库链接	231
第 10 章 事务、锁、闪回和 Undo 表空间	206	11.2.1 创建数据库链接	232
10.1 事务	206	11.2.2 使用数据库链接	233
10.1.1 事务的概念	206	11.2.3 删除数据库链接	234
10.1.2 事务处理	207	11.3 快照	234
10.1.3 自治事务	211	11.3.1 创建快照	234
10.2 锁	212	11.3.2 修改快照	238
10.2.1 锁机制和死锁	213	11.3.3 删除快照	239
10.2.2 锁的类型	213	11.4 序列	239
10.2.3 表锁和事务锁	214	11.4.1 创建序列	240
10.3 闪回操作	215	11.4.2 修改序列	241
		11.4.3 删除序列	242

第 2 部分 实 验

实验 1 创建数据库和表	243	实验 5 PL/SQL 编程	259
实验 2 表数据插入、修改和删除	246	实验 6 存储过程和触发器的使用	261
实验 3 数据库的查询和视图	249	实验 7 用户、角色与安全性	263
实验 4 索引和完整性	257		

第 3 部分 Oracle 12c 综合应用

实习 0 创建实习数据库	266	实习 1 PHP 5/Oracle 12c 学生成绩 管理系统	272
实习 0.1 创建数据库及其对象	266	实习 1.1 PHP 开发平台搭建	272
实习 0.2 功能和界面	269		

实习 1.1.1 创建 PHP 环境·····	272	实习 2.3.1 主界面·····	302
实习 1.1.2 Eclipse 安装与配置·····	275	实习 2.3.2 功能导航·····	303
实习 1.2 PHP 开发入门·····	277	实习 2.4 学生管理·····	305
实习 1.2.1 PHP 项目的建立·····	277	实习 2.4.1 界面设计·····	305
实习 1.2.2 PHP 项目的运行·····	278	实习 2.4.2 功能实现·····	308
实习 1.2.3 PHP 连接 Oracle 12c·····	279	实习 2.5 成绩管理·····	313
实习 1.3 系统主页设计·····	281	实习 2.5.1 界面设计·····	313
实习 1.3.1 主界面·····	281	实习 2.5.2 功能实现·····	316
实习 1.3.2 功能导航·····	282	实习 3 Visual C# 2013/Oracle 12c 学生	
实习 1.4 学生管理·····	283	成绩管理系统·····	321
实习 1.4.1 界面设计·····	283	实习 3.1 ADO.NET 架构原理·····	321
实习 1.4.2 功能实现·····	286	实习 3.2 创建 Visual C# 项目·····	322
实习 1.5 成绩管理·····	288	实习 3.2.1 Visual C# 项目的建立·····	322
实习 1.5.1 界面设计·····	288	实习 3.2.2 VS 2013 连接	
实习 1.5.2 功能实现·····	291	Oracle 12c·····	322
实习 2 Java EE 7/Oracle 12c 学生成绩		实习 3.3 系统主界面设计·····	324
管理系统·····	293	实习 3.3.1 总体布局·····	324
实习 2.1 Java EE 开发平台搭建·····	293	实习 3.3.2 详细设计·····	325
实习 2.1.1 安装软件·····	293	实习 3.4 学生管理·····	327
实习 2.1.2 环境整合·····	295	实习 3.4.1 程序主体结构·····	327
实习 2.2 创建 Struts 2 项目·····	297	实习 3.4.2 功能实现·····	328
实习 2.2.1 创建 Java EE 项目·····	297	实习 3.5 成绩管理·····	332
实习 2.2.2 加载 Struts 2 包·····	299	实习 3.5.1 课程名加载·····	332
实习 2.2.3 连接 Oracle 12c·····	300	实习 3.5.2 功能实现·····	334
实习 2.3 系统主页设计·····	302	附录 A XSCJ 数据库样本数据·····	338

第1部分 实用教程

第1章 数据库的基本概念

Oracle 是甲骨文公司开发的数据库管理系统，在介绍 Oracle 数据库之前，首先介绍数据库的基本概念。

1.1 数 据 库

1.1.1 数据库、数据库管理系统和数据库系统

1. 数据库

数据库 (DB) 是存放数据的仓库，只不过这些数据存在一定的关联，并按一定的格式存放在计算机上。从广义上讲，数据不仅包含数字，还包括了文本、图像、音频、视频等。

例如，把一个学校的学生、课程、学生成绩等数据有序地组织并存放在计算机内，就可以构成一个数据库。因此，数据库由一些持久的相互关联数据的集合组成，并以一定的组织形式存放在计算机的存储介质中。数据库是事务处理、信息管理等应用系统的基础。

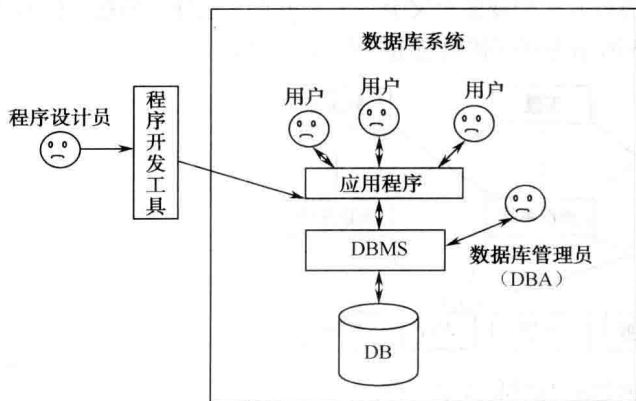


图 1.1 数据库系统的构成

成。

2. 数据库管理系统

数据库管理系统 (DBMS) 按一定的数据模型组织数据、管理数据库。数据库应用系统通过 DBMS 提供的接口操作数据库，数据库管理员 (DBA) 通过 DBMS 提供的界面管理、操作数据库。

数据、数据库、数据库管理系统与操作数据库的应用程序，加上支撑它们的硬件平台、软件平台和与数据库有关的人员一起构成了一个完整的数据库系统。如图 1.1 所示描述了数据库系统的构

1.1.2 数据模型

数据模型是指数据库管理系统中数据的存储结构，数据库管理系统根据数据模型对数据进行存储和管理，常见的数据模型有：层次模型、网状模型和关系模型。

1. 层次模型

层次模型是最早用于商品数据库管理系统的数据模型，它以树状层次结构组织数据。树形结构的每个节点表示一个记录类型，记录之间的联系是一对多的联系。位于树形结构顶部的节点称为根节点，层次模型有且仅有一个根节点。根节点以外的其他节点有且仅有一个父节点。图 1.2 所示为某学校按层次模型组织的数据示例。

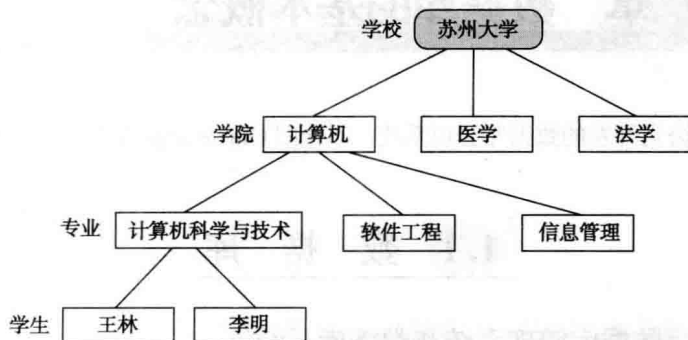


图 1.2 某学校按层次模型组织的数据示例

层次模型结构简单，容易实现，对于某些特定的应用系统效率很高，但在需要动态访问数据（如增加或修改记录类型）时，效率并不高。另外，对于一些非层次性结构（如多对多联系），层次模型表达起来比较烦琐和不直观。

2. 网状模型

网状模型可以看成层次模型的一种扩展。它采用网状结构组织数据，每个节点表示一个记录类型，记录之间的联系是一对多的联系。一个节点可以有一个或多个父节点和子节点，这样，数据库中的所有数据节点就构成了一个复杂的网络。图 1.3 所示为按网状模型组织的数据示例。

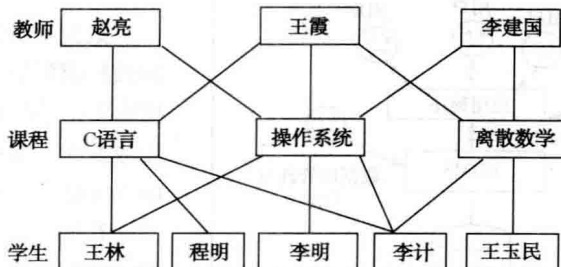


图 1.3 按网状模型组织的数据示例

与层次模型相比，网状模型具有更大的灵活性，更直接地描述现实世界，性能和效率也较好。网状模型的缺点是结构复杂，用户不易掌握。

3. 关系模型

关系模型是目前应用最多、最为重要的一种数据模型。关系模型建立在严格的数学概念基础上，以二维表格（关系表）的形式组织数据库中的数据，二维表由行和列组成。从用户观点看，关系模型是由一组关系组成的，关系之间通过公共属性产生联系。每个关系的数据结构是一个规范化的二维表，所以一个关系数据库就是由若干个表组成的。图 1.4 所示为按关系模型组织的数据示例。

在如图 1.4 所示的关系模型中，描述学生信息时使用的“学生”表涉及的主要信息有：学号、姓名、性别、出生时间、专业、总学分及备注。

表格中的一行称为一个记录，一列称为一个字段，每列的标题称为字段名。如果给关系表取一个名字，则有 n 个字段的关系表的结构可表示为：关系表名（字段名 1，…，字段名 n ），通常把关系表的结构称为关系模式。

在关系表中，如果一个字段或几个字段组合的值可唯一标识其对应记录，则称该字段或字段组合为码。例如，学生的“学号”可唯一标识每一个学生，则“学号”字段为“学生”表的码。有时，一个表可能有多个码，对于每一个关系表通常可指定一个码为“主码”，在关系模式中，一般用下划线标出主码。

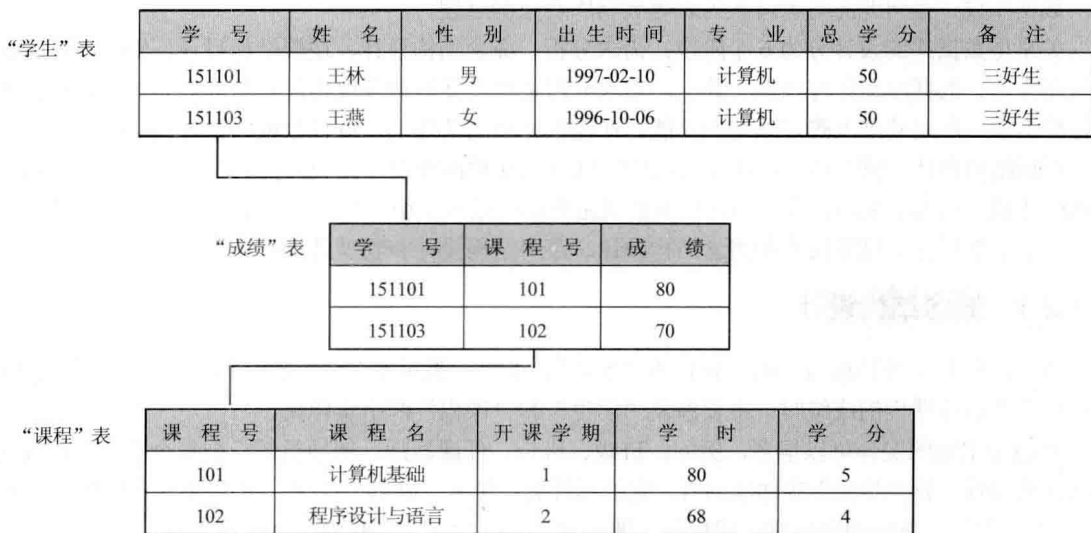


图 1.4 按关系模型组织的数据示例

设“学生”表的名字为 XSB，关系模式可表示为：XSB（学号，姓名，性别，出生时间，专业，总学分，备注）。

从图 1.4 可以看出，按关系模型组织数据表达方式简洁、直观，插入、删除、修改操作方便，而按层次、网状模型组织数据表达方式、操作比较复杂，因此，关系模型得到广泛应用，关系型数据库管理系统（RDBMS）成为主流。Oracle 数据库正是支持关系模型的数据库管理系统。

1.1.3 关系型数据库语言

SQL 语言是用于关系数据库查询的结构化语言，最早由 Boyce 和 Cha MBedin 在 1974 年提出。1976 年，SQL 开始在商品化关系数据库管理系统中应用。1982 年，美国标准化组织 ANSI 确认 SQL 为数据库系统的工业标准。1992 年，国际标准化组织（ISO）和国际电工委员会（IEC）发布了 SQL

国际标准,称为 SQL-92。

SQL 语言按功能可分为如下 3 部分:

(1) 数据定义语言 DDL (Data Definition Language): 定义数据库对象,包括定义表、视图和索引等。

(2) 数据操纵语言 DML (Data Manipulation Language): 主要对数据库中数据进行查询、插入、删除和修改操作。

(3) 数据控制语言 DCL (Data Control Language): 主要包括数据库的安全性控制、完整性控制,以及事务并发控制和故障恢复等语句。

目前,许多关系型数据库管理系统均支持 SQL 语言,如 Oracle、SQL Server、MySQL 等。但不同数据库管理系统之间的 SQL 语言不能完全通用。例如,甲骨文公司的 Oracle 数据库所使用的 SQL 语言是 Procedural Language/SQL (简称 PL/SQL),而微软公司的 SQL Server 数据库系统支持的则是 Transact-SQL (简称 T-SQL)。

1.2 数据库设计

数据库设计是将业务对象转换为表等数据库对象的过程。

关系型数据库的设计分为 6 个阶段:需求分析、概念结构设计、逻辑结构设计、物理结构设计、数据库实施、数据库运行与维护。其中,需求分析是通过详细调查现实世界要处理的对象,明确用户的各种需求,在此基础上确定系统的功能。在需求分析的基础上,进行数据库设计,包括概念结构设计、逻辑结构设计、物理结构设计。数据库实施就是在数据库设计完成后由数据库管理员 (DBA) 在 DBMS 上进行操作,完成设计。运行与维护就是数据库投入使用,DBA 对数据库进行管理、维护和处置,保证正常运行。这里仅介绍概念结构设计、逻辑结构设计和物理结构设计。

1.2.1 概念结构设计

通常,把每一类数据对象的个体称为“实体”,而每一类对象个体的集合称为“实体集”。例如,在管理学生所选课程的成绩时,主要涉及“学生”和“课程”两个实体集。

其他非主要的实体可以很多,例如,班级、班长、任课教师、辅导员等。把每个实体集涉及的信息项称为属性。就“学生”实体集而言,它的属性有:学号、姓名、性别、出生时间、专业、总学分、备注;“课程”实体集的属性有:课程号、课程名、开课学期、学时和学分。

实体集“学生”和实体集“课程”之间存在“选课”的关系,通常把这类关系称为“联系”,将实体集及实体集联系的图称为 E-R 模型。E-R 模型的表示方法为:

- (1) 实体集采用矩形框表示,框内为实体名;
- (2) 实体的属性采用椭圆框表示,框内为属性名,并用无向边与其相应实体集连接;
- (3) 实体间的联系采用菱形框表示,联系以适当的含义命名,名字写在菱形框中,用无向边将参加联系的实体矩形框分别与菱形框相连,并在连线上标明联系的类型,即 1—1、1—n 或 m—n;
- (4) 如果一个联系有属性,则这些属性也应采用无向边与该联系相连接起来。

因此,E-R 模型也称为 E-R 图。关系数据库的设计者通常使用 E-R 图来对信息世界建模。从分析用户项目涉及的数据对象及数据对象之间的联系出发,到获取 E-R 图的这一过程就称为概念结构设计。

两个实体集 A 和 B 之间的联系可能是以下 3 种情况之一。

1. 一对一的联系 (1:1)

A 中的一个实体至多与 B 中的一个实体相联系, B 中的一个实体也至多与 A 中的一个实体相联系。例如, “班级”与“正班长”这两个实体集之间的联系是一对一的联系, 因为一个班只有一个正班长, 反过来, 一个正班长只属于一个班。“班级”与“正班长”两个实体集的 E-R 模型如图 1.5 所示。

2. 一对多的联系 (1:n)

A 中的一个实体可以与 B 中的多个实体相联系, 而 B 中的一个实体至多与 A 中的一个实体相联系。例如, “班级”与“学生”这两个实体集之间的联系是一对多的联系, 因为一个班可有若干学生, 反过来, 一个学生只能属于一个班。“班级”与“学生”两个实体集的 E-R 模型如图 1.6 所示。

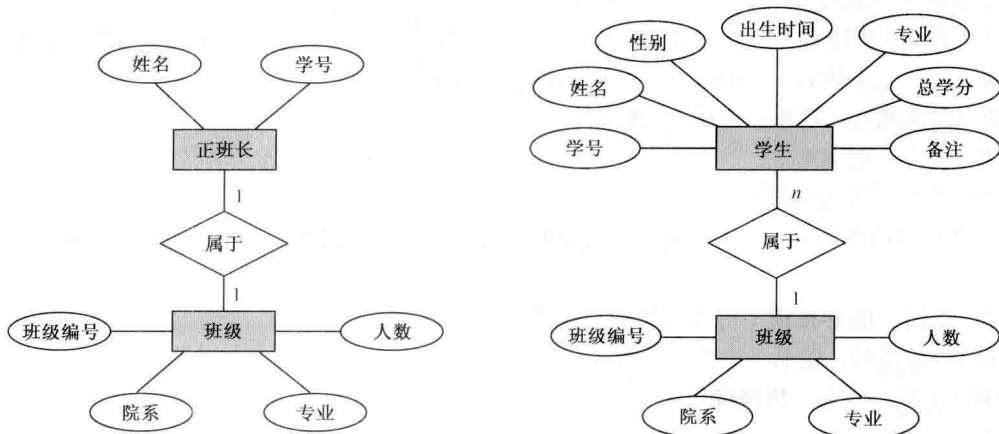


图 1.5 “班级”与“正班长”两个实体集的 E-R 模型 图 1.6 “班级”与“学生”两个实体集的 E-R 模型

3. 多对多的联系 (m:n)

A 中的一个实体可以与 B 中的多个实体相联系, 而 B 中的一个实体也可与 A 中的多个实体相联系。例如, “学生”与“课程”这两个实体集之间的联系是多对多的联系, 因为, 一个学生可选多门课程, 反过来, 一门课程可被多个学生选修, 每个学生选修了一门课以后都有一个成绩。“学生”与“课程”两个实体集的 E-R 模型如图 1.7 所示。

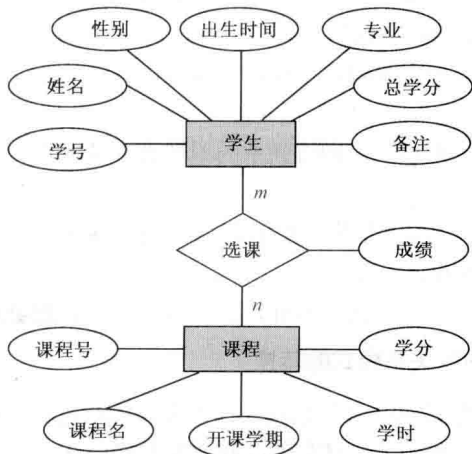


图 1.7 “学生”与“课程”两个实体集的 E-R 模型

1.2.2 逻辑结构设计

用 E-R 图描述学生成绩管理系统中实体集与实体集之间的联系,目的是以 E-R 图为工具设计出关系模式,即确定应用系统所使用的数据库应包含的表和表的结构。通常,这一设计过程就称为逻辑结构设计。

1. (1:1) 联系的 E-R 图到关系模式的转换

对于 (1:1) 的联系既可单独对应一个关系模式,也可以不单独对应一个关系模式。

(1) 联系单独对应一个关系模式,则由联系属性、参与联系的各实体集的主码属性构成关系模式,其主码可选参与联系的实体集的任一方的主码。

例如,图 1.5 描述的“班级(BJB)”与“正班长(BZB)”实体集通过“属于(SYB)”联系的 E-R 模型可设计如下关系模式(下画线“ ”表示该字段为主码):

BJB (班级编号, 院系, 专业, 人数)

BZB (学号, 姓名)

SYB (学号, 班级编号)

(2) 联系不单独对应一个关系模式,联系的属性及一方的主码加入另一方实体集对应的关系模式中。

例如,图 1.5 的 E-R 模型可设计如下关系模式:

BJB (班级编号, 院系, 专业, 人数)

BZB (学号, 姓名, 班级编号)

或者

BJB (班级编号, 院系, 专业, 人数, 学号)

BZB (学号, 姓名)

2. (1:n) 联系的 E-R 图到关系模式的转换

对于 (1:n) 的联系既可单独对应一个关系模式,也可以不单独对应一个关系模式。

(1) 联系单独对应一个关系模式,则由联系的属性、参与联系的各实体集的主码属性构成关系模式, n 端的主码作为该关系模式的主码。

例如,图 1.6 描述的“班级(BJB)”与“学生(XSB)”实体集的 E-R 模型可设计如下关系模式:

BJB (班级编号, 院系, 专业, 人数)

XSB (学号, 姓名, 性别, 出生时间, 专业, 总学分, 备注)

SYB (学号, 班级编号)

(2) 联系不单独对应一个关系模式,则将联系的属性及 1 端的主码加入 n 端实体集对应的关系模式中,主码仍为 n 端的主码。

例如,图 1.6 “班级(BJB)”与“学生(XSB)”实体集 E-R 模型可设计如下关系模式:

BJB (班级编号, 院系, 专业, 人数)

XSB (学号, 姓名, 性别, 出生时间, 专业, 总学分, 备注, 班级编号)

3. (m:n) 联系的 E-R 图到关系模式的转换

对于 (m:n) 的联系,单独对应一个关系模式,该关系模式包括联系的属性、参与联系的各实体集的主码属性,该关系模式的主码由各实体集的主码属性共同组成。

例如,图 1.7 描述的“学生(XSB)”与“课程(KCB)”实体集之间的联系可设计如下关系模式:

XSB (学号, 姓名, 性别, 出生时间, 专业, 总学分, 备注)

KCB (课程号, 课程名称, 开课学期, 学时, 学分)

CJB (学号, 课程号, 成绩)

关系模式 CJB 的主码是由“学号”和“课程号”两个属性组合构成的, 一个关系模式只能有一个主码。

1.2.3 物理结构设计

数据库在物理设备上的存储结构与存取方法称为数据库的物理结构。数据库的物理结构设计通常分为 2 步:

- (1) 确定数据库的物理结构, 在关系数据库中主要指存取方法和存储结构;
- (2) 对物理结构进行评价, 评价的重点是时间和空间效率。

1.3 数据库应用系统

数据库应用系统可以采用客户-服务器 (C/S) 模式应用系统和三层客户-服务器 (B/S) 模式应用系统。

1.3.1 C/S 模式应用系统

客户-服务器 (C/S) 模式应用系统操作数据库方式如图 1.8 所示。由于应用程序直接与用户打交道, 而数据库管理系统不直接与用户打交道, 所以应用程序被称为“前台”, 而数据库管理系统被称为“后台”。由于应用程序向数据库管理系统提出服务请求, 故通常称为客户程序 (Client); 而数据库管理系统为应用程序提供服务, 故通常称为服务器程序 (Server), 又将这种操作数据库模式称为客户-服务器 (C/S) 模式。

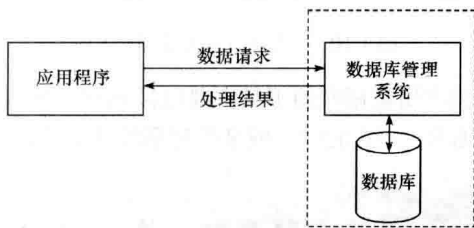


图 1.8 客户-服务器 (C/S) 模式应用系统操作数据库方式

目前, 开发数据库应用系统比较流行的工具包括 Visual C#、Visual Basic、Visual C++、PowerBuilder 等。

应用程序和数据库管理系统可以运行在同一台计算机上 (单机方式), 也可以运行在网络方式下。在网络方式下, 数据库管理系统在网络上的一台主机上运行, 应用程序可以在网络上的多台主机上运行, 即一对多的方式。例如, 用 Visual C# 2013 开发的客户-服务器 (C/S) 模式学生成绩管理系统的学生管理界面如图 1.9 所示。



图 1.9 C/S 模式学生成绩管理系统的学生管理界面

1.3.2 B/S 模式应用系统

基于 Web 的数据库应用采用三层客户-服务器模式，也称为 B/S 结构。第一层为浏览器，第二层为 Web 服务器，第三层为数据库服务器。浏览器是用户输入数据和显示结果的交互界面。用户在浏览器表单中输入数据，然后将表单中的数据提交并发送到 Web 服务器。Web 服务器应用程序接收并处理用户的数据，通过数据库服务器，从数据库中查询需要的数据并返回给 Web 服务器。Web 服务器再把返回的结果插入 HTML 页面，传送到客户端，在浏览器中显示出来。三层客户-服务器结构如图 1.10 所示。

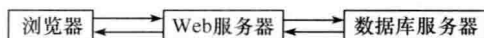


图 1.10 三层客户-服务器结构

目前，流行的开发 B/S 结构应用数据库的工具主要包括 PHP、Java EE、ASP.NET 等。例如，用 PHP 开发的三层客户-服务器（B/S）模式的学生成绩管理系统的学生管理页面如图 1.11 所示。



图 1.11 B/S 模式的学生成绩管理系统的学生管理页面

第 2 章 Oracle 12c 数据库介绍

Oracle 是目前世界上使用最为广泛的关系型数据库管理系统，在数据库领域一直处于领先地位，在信息系统管理、企业数据处理、电子商务、网络、大数据和云计算等各个方面广泛应用，最新版本为 Oracle 12c。

2.1 Oracle 简介与安装

2.1.1 Oracle 数据库简介

1979 年，甲骨文公司推出当时世界上第一个基于 SQL 标准的关系数据库产品 Oracle，之后不断完善，推出各种不同版本以适应新的应用需要。从 Oracle 2.0 开始一直到 Oracle 7.3.4 都只是简单的版本号，但从 Oracle 8 开始就出现数据库产品特性标识符，如 Oracle 8i 和 9i，这个 i 是 Internet 的缩写，表示该产品全面支持 Internet 应用。从 Oracle 10 开始称为 10g，这个 g 是 Grid 的缩写，表示支持“网络运算”，就是更好地支持集群和多点应用。Oracle 11 包括 11g 和 11i，Oracle 11g 是在 Oracle 10g 的基础上对功能进行了增强。Oracle 11i 的全称是 Oracle Applications 11i，这里的 i 是 Intelligence 的缩写，它不是数据库产品，而是一套电子商务套件。

随着互联网普及，甲骨文公司的产品发展战略也转而针对网络和云，进行了全面的升级。2013 年 6 月，甲骨文正式发布 Oracle 12c (12.1.0.1.0)，作为首款面向云计算的数据库产品，它的整个研发过程投入了大量人力和资源，包括长达 5 年的研发投入、2500 多人年的工作量、3000 多个系统的功能测试和 120 多万小时的压力测试。Oracle 12c 在前版 Oracle 11g 的基础上增加了 500 多种新特性，不仅包含更快的运行性能、简化的云端数据库整合、面向海量数据的分析与存储以及企业级的安全防护，而且提供了独具特色的多租户架构。

2014 年 8 月，甲骨文又发布了 Oracle 12c 的更新版(12.1.0.2.0)，该版包含 Oracle Database In-Memory (数据库内存选件)、Oracle 大数据 SQL 以及通过 SQL/REST 接口对 JSON 数据进行查询等新功能，使 Oracle 12c 成为私有云和公有云部署的理想平台。

可见，Oracle 12c 是一个重量级的 DBMS 系统软件，其所囊括的技术知识是极其庞大复杂的！作为教材，本书只能讲述它的基本功能和应用，并不涉及上述多租户、大数据、云等这些概念和内容（普通高校也不具备这样的实践条件），对此感兴趣的读者可去看这方面的专业技术书，或在从事与 Oracle 有关的工作后再继续深造。

2.1.2 Oracle 12c 的安装

1. 安装前的准备

因 Oracle 是企业重量级软件，对计算机系统的配置要求很高，Oracle 12c 只能在 64 位 Windows 平台上运行，故读者事先要准备一台预装了 64 位 Windows 操作系统的计算机。若配置不给力还要升级硬件，则建议采用多核处理器和大容量（至少 1G 以上）的内存。本书笔者的计算机操作系统是