



工业和信息化普通高等教育“十二五”规划教材

21世纪高等教育计算机规划教材

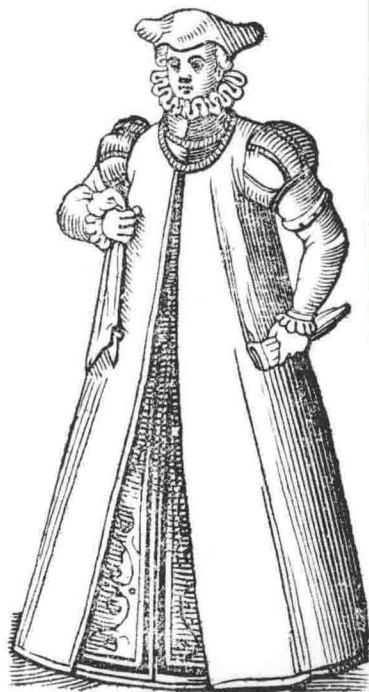
COMPUTER

# MySQL 数据库基础 与实例教程

MySQL Fundamentals & Practices

孔祥盛 主编

- 入门门槛低，讲解细腻，配套资源丰富、完善
- 案例虽小，五脏俱全，10张数据库表囊括 MySQL 所有知识点
- 强调实训环节与 MySQL 知识的完美融合



 人民邮电出版社  
POSTS & TELECOM PRESS



工业和信息化普通高等教育“十二五”规划教材

21世纪高等教育计算机规划教材

COMPUTER

# MySQL 数据库基础 与实例教程

MySQL Fundamentals & Practices

■ 孔祥盛 主编



人民邮电出版社

北京

## 图书在版编目 (C I P) 数据

MySQL数据库基础与实例教程 / 孔祥盛主编. -- 北京: 人民邮电出版社, 2014. 6  
21世纪高等教育计算机规划教材  
ISBN 978-7-115-35338-2

I. ①M… II. ①孔… 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(2014)第075998号

## 内 容 提 要

作为世界上最受欢迎的开源数据库之一, MySQL 由于其性能优越、功能强大, 受到了广大自由软件爱好者甚至是商业软件用户的青睐。本书以讲解 MySQL 基础知识为目标, 以案例的实现为载体, 以不同的章节完成不同的任务为理念, 深入讲解关系数据库设计、MySQL 基础知识以及 MySQL 编程知识。

本书内容丰富、讲解深入, 适用于初学者快速上手, 特别适合用作各类院校相关专业的教材。本书入门门槛低, 非常适合作为培训机构的培训教材, 也适用于计算机二级考试 MySQL 数据库程序设计的培训教材, 同时也是一本面向广大 MySQL 爱好者的实用参考书。

- 
- ◆ 主 编 孔祥盛  
责任编辑 李海涛  
责任印制 彭志环 杨林杰
  - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号  
邮编 100164 电子邮件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
北京中新伟业印刷有限公司印刷
  - ◆ 开本: 787×1092 1/16  
印张: 20.5 2014年6月第1版  
字数: 539千字 2014年6月北京第1次印刷
- 

定价: 45.00 元

读者服务热线: (010)81055256 印装质量热线: (010)81055316  
反盗版热线: (010)81055315



### 3. 选用尽可能少的数据库表讲解 MySQL 尽可能多的知识

本书选用尽可能少的数据库表讲解 MySQL 尽可能多的知识,以便读者能够将所有精力集中在 MySQL 知识点的学习上。本书所使用的数据库表不超过 10 张,经常使用的数据库表不超过 5 张,使用 5 张表讲解 MySQL 几乎所有的知识点,很大程度上可以减轻读者的负担。

### 4. 内容丰富、严谨

本书内容丰富,几乎囊括 MySQL 所有核心知识点。本书的内容编排一气呵成,且遵循知识的学习曲线,并尽量做到不留死角。本书对 MySQL 内容的选取非常严谨,一环扣一环,从一个知识点过渡到另一个知识点非常顺畅和自然,章节之间循序渐进,内容不冲突、不重复、不矛盾。本书不仅是一本介绍 MySQL 的书籍,更是一本介绍数据库技术的书籍。

### 5. 注重软件工程在数据库开发过程中的应用

数据库初学者通常存在致命的误区:重开发,轻设计。开发出来的数据库往往成了倒立的金字塔,头重脚轻。真正的数据库开发,首先强调的是设计,其次是开发。正因为如此,本书将软件工程的思想融入到数据库开发过程中,并对数据库设计的相关知识,在本书第 1 章中进行了详细讲解。

### 6. 强调实训环节与 MySQL 知识的完美融合

数据库技术用于解决商业领域的商业问题,本书提供的综合实训做到了项目实训与 MySQL 知识的完美融合。通过项目实训,读者可以更清楚地了解应用程序的开发流程以及数据库技术在应用程序中举足轻重的地位。

### 7. 配套资源丰富、完善

为方便教师教学、读者自学,本书提供的配套资源包括:PPT 电子教案、所有示例程序源代码以及本书涉及的所有软件安装程序。读者可以在“人民邮电出版社教学服务与资源网(<http://www.ptpedu.com/>)”上免费下载本书配套资源。

鉴于上述特点,笔者相信:本书能够成为一本教师教学、学生自学的好“剧本”,是面向广大 MySQL 爱好者的实用参考书。

本书由孔祥盛任主编,张永华、王珍、侯国平、赵春霞、孙大鹏、刘炜和王重英任副主编。参加编写的人员还有孙婧和王娜。其中刘炜编写第 1 章和第 2 章,赵春霞编写第 3 章,王重英编写第 4 章和第 6 章,孙大鹏编写第 5 章,王珍编写第 7 章和第 8 章,张永华编写第 9 章,侯国平编写第 10 章,孙婧、王娜对本书综合实训的代码进行了编写,并对全书的代码进行了测试,孔祥盛对本书的案例以及组织架构进行了设计。全书由孙婧、王娜预审,孔祥盛统稿审定。

由于本书涉及面广,加之笔者经验有限,书中难免存在不妥之处,敬请广大读者批评指正。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。

编 者

2014 年 2 月

# 目 录

## 第一篇 关系数据库设计

### 第 1 章 数据库设计概述 ..... 2

#### 1.1 数据库概述 ..... 2

##### 1.1.1 关系数据库管理系统 ..... 2

##### 1.1.2 关系数据库 ..... 3

##### 1.1.3 结构化查询语言 SQL ..... 4

#### 1.2 数据库设计的相关知识 ..... 5

##### 1.2.1 商业知识和沟通技能 ..... 5

##### 1.2.2 数据库设计辅助工具 ..... 6

##### 1.2.3 “选课系统”概述 ..... 7

##### 1.2.4 定义问题域 ..... 8

##### 1.2.5 编码规范 ..... 8

#### 1.3 E-R 图 ..... 9

##### 1.3.1 实体和属性 ..... 10

##### 1.3.2 关系 ..... 10

##### 1.3.3 E-R 图的设计原则 ..... 12

#### 1.4 关系数据库设计 ..... 15

##### 1.4.1 为每个实体建立一张数据库表 ..... 15

##### 1.4.2 为每张表定义一个主键 ..... 16

##### 1.4.3 增加外键表示一对多关系 ..... 17

##### 1.4.4 建立新表表示多对多关系 ..... 19

##### 1.4.5 为字段选择合适的数据类型 ..... 20

##### 1.4.6 定义约束 (constraint) 条件 ..... 20

##### 1.4.7 评价数据库表设计的质量 ..... 22

##### 1.4.8 使用规范化减少数据冗余 ..... 22

##### 1.4.9 避免数据经常发生变化 ..... 26

#### 习题 ..... 28

## 第二篇 MySQL 基础

### 第 2 章 MySQL 基础知识 ..... 30

#### 2.1 MySQL 概述 ..... 30

##### 2.1.1 MySQL 的特点 ..... 30

##### 2.1.2 MySQL 服务的安装 ..... 32

##### 2.1.3 MySQL 服务的配置 ..... 35

##### 2.1.4 启动与停止 MySQL 服务 ..... 40

##### 2.1.5 MySQL 配置文件 ..... 41

##### 2.1.6 MySQL 客户机 ..... 42

##### 2.1.7 连接 MySQL 服务器 ..... 43

#### 2.2 字符集以及字符序设置 ..... 45

##### 2.2.1 字符集及字符序概念 ..... 45

##### 2.2.2 MySQL 字符集与字符序 ..... 45

##### 2.2.3 MySQL 字符集的转换过程 ..... 47

##### 2.2.4 MySQL 字符集的设置 ..... 48

##### 2.2.5 SQL 脚本文件 ..... 49

#### 2.3 MySQL 数据库管理 ..... 50

##### 2.3.1 创建数据库 ..... 50

##### 2.3.2 查看数据库 ..... 51

##### 2.3.3 显示数据库结构 ..... 52

##### 2.3.4 选择当前操作的数据库 ..... 52

##### 2.3.5 删除数据库 ..... 52

#### 2.4 MySQL 表管理 ..... 53

##### 2.4.1 MyISAM 和 InnoDB 存储引擎 ..... 53

##### 2.4.2 设置默认的存储引擎 ..... 54

##### 2.4.3 创建数据库表 ..... 54

##### 2.4.4 显示表结构 ..... 55

##### 2.4.5 表记录的管理 ..... 56

##### 2.4.6 InnoDB 表空间 ..... 59

##### 2.4.7 删除表 ..... 61

#### 2.5 系统变量 ..... 61

##### 2.5.1 全局系统变量与会话系统变量 ..... 61

##### 2.5.2 查看系统变量的值 ..... 62

##### 2.5.3 设置系统变量的值 ..... 64

#### 2.6 MySQL 数据库备份和恢复 ..... 65

习题 .....	67	4.3 表记录的删除 .....	103
<b>第 3 章 MySQL 表结构的管理</b> .....	<b>68</b>	4.3.1 使用 delete 删除表记录 .....	103
3.1 MySQL 数据类型 .....	68	4.3.2 使用 truncate 清空表记录 .....	104
3.1.1 MySQL 整数类型 .....	68	4.4 MySQL 特殊字符序列 .....	106
3.1.2 MySQL 小数类型 .....	69	习题 .....	108
3.1.3 MySQL 字符串类型 .....	70	<b>第 5 章 表记录的检索</b> .....	<b>109</b>
3.1.4 MySQL 日期类型 .....	71	5.1 select 语句概述 .....	109
3.1.5 MySQL 复合类型 .....	73	5.1.1 使用 select 子句指定字段列表 .....	110
3.1.6 MySQL 二进制类型 .....	75	5.1.2 使用谓词过滤记录 .....	111
3.1.7 选择合适的数据类型 .....	75	5.1.3 使用 from 子句指定数据源 .....	112
3.2 创建表 .....	76	5.1.4 多表连接 .....	115
3.2.1 设置约束 .....	77	5.2 使用 where 子句过滤结果集 .....	116
3.2.2 设置自增型字段 .....	81	5.2.1 使用单一的条件过滤结果集 .....	116
3.2.3 其他选项的设置 .....	81	5.2.2 is NULL 运算符 .....	117
3.2.4 创建“选课系统”数据库表 .....	82	5.2.3 select 语句与字符集 .....	118
3.2.5 复制一个表结构 .....	83	5.2.4 使用逻辑运算符 .....	119
3.3 修改表结构 .....	84	5.2.5 使用 like 进行模糊查询 .....	121
3.3.1 修改字段相关信息 .....	84	5.3 使用 order by 子句对结果集排序 .....	122
3.3.2 修改约束条件 .....	85	5.4 使用聚合函数汇总结果集 .....	123
3.3.3 修改表的其他选项 .....	87	5.5 使用 group by 子句对记录分组	
3.3.4 修改表名 .....	87	统计 .....	125
3.4 删除表 .....	87	5.5.1 group by 子句与聚合函数 .....	125
3.5 索引 .....	88	5.5.2 group by 子句与 having 子句 .....	126
3.5.1 理解索引 .....	88	5.5.3 group by 子句与 group_concat()	
3.5.2 索引关键字的选取原则 .....	91	函数 .....	127
3.5.3 索引与约束 .....	92	5.5.4 group by 子句与 with rollup	
3.5.4 创建索引 .....	93	选项 .....	127
3.5.5 删除索引 .....	94	5.6 合并结果集 .....	128
习题 .....	95	5.7 子查询 .....	129
<b>第 4 章 表记录的更新操作</b> .....	<b>96</b>	5.7.1 子查询与比较运算符 .....	129
4.1 表记录的插入 .....	96	5.7.2 子查询与 in 运算符 .....	131
4.1.1 使用 insert 语句插入新记录 .....	96	5.7.3 子查询与 exists 逻辑运算符 .....	132
4.1.2 更新操作与字符集 .....	99	5.7.4 子查询与 any 运算符 .....	133
4.1.3 关于自增型字段 .....	100	5.7.5 子查询与 all 运算符 .....	133
4.1.4 批量插入多条记录 .....	100	5.8 选课系统综合查询 .....	134
4.1.5 使用 insert...select 插入结果集 .....	101	5.9 使用正则表达式模糊查询 .....	138
4.1.6 使用 replace 插入新记录 .....	102	5.10 全文检索 .....	139
4.2 表记录的修改 .....	103	5.10.1 全文检索的简单应用 .....	140
		5.10.2 全文检索方式 .....	144

5.10.3 布尔检索模式的复杂应用	144	7.2 触发器	201
5.10.4 MySQL 全文检索的注意事项	146	7.2.1 准备工作	202
5.10.5 InnoDB 表的全文检索	146	7.2.2 使用触发器实现检查约束	203
习题	147	7.2.3 使用触发器维护冗余数据	204
<b>第三篇 MySQL 编程</b>		7.2.4 使用触发器模拟外键级联选项	205
<b>第 6 章 MySQL 编程基础</b>		7.2.5 查看触发器的定义	206
6.1 MySQL 编程基础知识	150	7.2.6 删除触发器	207
6.1.1 常量	150	7.2.7 使用触发器的注意事项	207
6.1.2 用户自定义变量	152	7.3 临时表	208
6.1.3 运算符与表达式	157	7.3.1 临时表概述	208
6.1.4 begin-end 语句块	159	7.3.2 临时表的创建、查看与删除	208
6.1.5 重置命令结束标记	160	7.3.3 “选课系统”中临时表的使用	209
6.2 自定义函数	160	7.3.4 使用临时表的注意事项	210
6.2.1 创建自定义函数的语法格式	160	7.4 派生表 (derived table)	211
6.2.2 函数的创建与调用	161	7.5 子查询、视图、临时表、派生表	211
6.2.3 函数的维护	163	习题	213
6.2.4 条件控制语句	165	<b>第 8 章 存储过程与游标</b>	214
6.2.5 循环语句	167	8.1 存储过程	214
6.3 系统函数	171	8.1.1 创建存储过程的语法格式	214
6.3.1 数学函数	171	8.1.2 存储过程的调用	215
6.3.2 字符串函数	173	8.1.3 “选课系统”的存储过程	216
6.3.3 数据类型转换函数	180	8.1.4 查看存储过程的定义	220
6.3.4 条件控制函数	180	8.1.5 删除存储过程	221
6.3.5 系统信息函数	182	8.1.6 存储过程与函数的比较	222
6.3.6 日期和时间函数	183	8.2 错误触发条件和错误处理	223
6.3.7 其他常用的 MySQL 函数	189	8.2.1 自定义错误处理程序	223
6.4 中文全文检索的模拟实现	191	8.2.2 自定义错误触发条件	225
习题	193	8.2.3 自定义错误处理程序说明	226
<b>第 7 章 视图与触发器</b>	195	8.3 游标	226
7.1 视图	195	8.3.1 使用游标	226
7.1.1 创建视图	195	8.3.2 游标在“选课系统”中的使用	227
7.1.2 查看视图的定义	196	8.4 预处理 SQL 语句	229
7.1.3 视图在“选课系统”中的应用	197	8.4.1 预处理 SQL 语句使用步骤	229
7.1.4 视图的作用	198	8.4.2 “选课系统”中预处理 SQL 语句的 使用	230
7.1.5 删除视图	199	8.4.3 预处理 SQL 语句的复杂应用	231
7.1.6 检查视图	199	8.4.4 静态 SQL 语句与预处理 SQL 语句	233
7.1.7 local 与 cascade 检查视图	200	8.5 存储程序的说明	234



习题 .....	234	10.3 网上选课系统的系统规划 .....	283
<b>第 9 章 事务机制与锁机制</b> .....	<b>235</b>	10.3.1 网上选课系统的目标 .....	284
9.1 事务机制 .....	235	10.3.2 网上选课系统的可行性分析 .....	284
9.1.1 事务机制的必要性 .....	235	10.3.3 网上选课系统的项目进度表 .....	284
9.1.2 关闭 MySQL 自动提交 .....	237	10.3.4 网上选课系统的人员分工 .....	285
9.1.3 回滚 .....	237	10.4 网上选课系统的系统分析 .....	286
9.1.4 提交 .....	239	10.4.1 网上选课系统的功能需求分析 .....	286
9.1.5 事务 .....	240	10.4.2 网上选课系统的非功能需求 分析 .....	288
9.1.6 保存点 .....	241	10.5 网上选课系统的系统设计 .....	288
9.1.7 “选课系统”中的事务 .....	243	10.6 网上选课系统的系统实施 .....	290
9.2 锁机制 .....	246	10.6.1 准备工作 .....	290
9.2.1 锁机制的必要性 .....	246	10.6.2 制作 PHP 连接 MySQL 服务器 函数 .....	291
9.2.2 MySQL 锁机制的基础知识 .....	248	10.6.3 制作 PHP 权限系统函数 .....	293
9.2.3 MyISAM 表的表级锁 .....	250	10.6.4 首页 index.php 的开发 .....	294
9.2.4 InnoDB 表的行级锁 .....	253	10.6.5 教师注册模块的开发 .....	295
9.2.5 “选课系统”中的行级锁 .....	255	10.6.6 登录模块的开发 .....	298
9.2.6 InnoDB 表的意向锁 .....	257	10.6.7 注销模块的开发 .....	300
9.2.7 InnoDB 行级锁与索引之间的关系 .....	258	10.6.8 添加班级模块的开发 .....	300
9.2.8 间隙锁与死锁 .....	261	10.6.9 学生注册模块的开发 .....	301
9.2.9 死锁与锁等待 .....	262	10.6.10 密码重置模块 .....	303
9.3 事务的 ACID 特性 .....	264	10.6.11 申报课程模块 .....	304
9.3.1 事务的 ACID 特性 .....	264	10.6.12 课程列表显示模块 .....	305
9.3.2 事务的隔离级别与并发问题 .....	265	10.6.13 审核申报课程 .....	307
9.3.3 设置事务的隔离级别 .....	266	10.6.14 取消已审核课程 .....	308
9.3.4 使用间隙锁避免幻读现象 .....	271	10.6.15 浏览自己申报的课程 .....	309
9.4 事务与锁机制注意事项 .....	272	10.6.16 删除课程 .....	310
习题 .....	273	10.6.17 学生选修或者调换已经审核的 课程 .....	310
<b>第四篇 综合实训</b>		10.6.18 查看自己选修的课程 .....	311
<b>第 10 章 网上选课系统的开发</b> .....	<b>276</b>	10.6.19 取消选修课程 .....	312
10.1 PHP 预备知识 .....	276	10.6.20 查看课程的学生信息列表 .....	313
10.1.1 为何选用 B/S 结构以及 PHP 脚本 语言 .....	276	10.6.21 查看选修人数少于 30 人的 课程信息 .....	314
10.1.2 PHP 脚本语言概述 .....	277	10.7 界面设计与 MVC 模式 .....	315
10.1.3 PHP 脚本程序的工作流程 .....	278	10.8 网上选课系统的测试 .....	317
10.1.4 Web 服务器的部署 .....	280	习题 .....	318
10.1.5 注意事项 .....	282	<b>参考文献</b> .....	<b>319</b>
10.2 软件开发生命周期 SDLC .....	283		

# 第一篇

# 关系数据库设计

数据库设计概述

数据库概述

数据库设计的相关知识

E-R 图

关系数据库设计

# 第 1 章

## 数据库设计概述

数据库是信息系统的核心，在信息社会中占据着举足轻重的地位。数据库技术主要研究如何科学地组织、存储和管理数据库中的数据。采用科学的方法开发、设计一个结构良好的数据库，是所有数据库开发人员应该掌握的最基本技能。本章抛开 MySQL 讲解关系数据库设计的相关知识，以“选课系统”为例，讲解“选课系统”数据库的设计流程。通过本章的学习，读者将具备一定的数据库设计能力。

### 1.1 数据库概述

简单地说，数据库（Database，DB）是存储、管理数据的容器；严格地说，数据库是“按照某种数据结构对数据进行组织、存储和管理的容器”。无论哪一种说法，数据永远是数据库的核心。

#### 1.1.1 关系数据库管理系统

数据是数据库的核心。数据库容器通常包含诸多数据库对象，如表、视图、索引、函数、存储过程、触发器、事件等，这些数据库对象最终都是以文件的形式存储在外存（例如硬盘）上。数据库用户如何能够访问到数据库容器中的数据库对象呢？事实上，通过“数据库管理系统”，数据库用户可以轻松地实现数据库容器中各种数据库对象的访问（增、删、改、查等操作），并可以轻松地完成数据库的维护工作（备份、恢复、修复等操作），如图 1-1 所示。

数据库管理系统（Database Management System，DBMS）安装于操作系统之上，是一个管理、控制数据库容器中各种数据库对象的系统软件。可以这样理解：数据库用户无法直接通过操作系统获取数据库文件中的具体内容；数据库管理系统通过调用操作系统的进程管理、内存管理、设备管理以及文件管理等服务，为数据库用

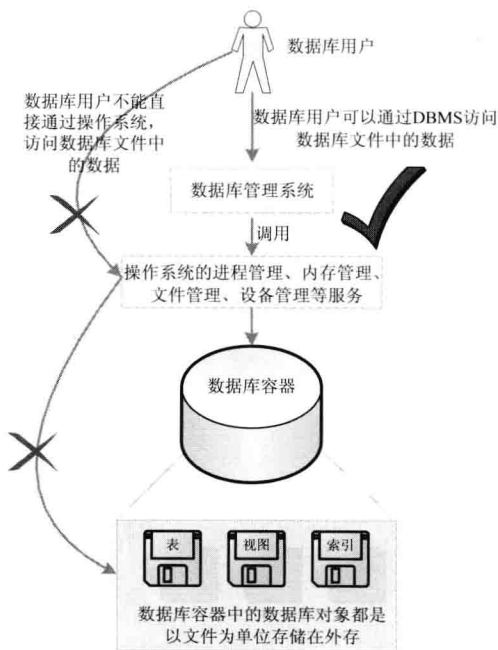


图 1-1 数据库管理系统与操作系统之间的关系

户提供管理、控制数据库容器中各种数据库对象、数据库文件的接口。

数据库管理系统通常会选择某种“数学模型”存储、组织、管理数据库中的数据，常用的数学模型包括“层次模型”、“网状模型”、“关系模型”以及“面向对象模型”等。基于“关系模型”的数据库管理系统称为关系数据库管理系统（Relational Database Management System, RDBMS）。随着关系数据库管理系统的日臻完善，目前关系数据库管理系统已占据主导地位。

通过关系数据库管理系统，数据库开发人员可以轻而易举地创建关系数据库容器，并在该数据库容器中创建各种数据库对象（表、索引、视图、存储过程、触发器、函数等）以及维护各种数据库对象。目前成熟的关系数据库管理系统主要源自欧美数据库厂商，典型的有美国微软公司的 SQL Server、美国 IBM 公司的 DB2 和 Informix、德国 SAP 公司的 Sybase、美国甲骨文公司的 Oracle，然而这些数据库都是商业数据库，且价格昂贵。截至目前，国产数据库管理系统还不成熟，国内很多大型企业，也不乏核心部门（例如金融银行、移动通信、石油行业、12306 铁路订票等）不得不依赖进口，“享受着”国外数据库厂商提供服务的同时，不得不将核心部门的信息交由国外数据库管理，企业的核心数据、核心部门的机密数据的不安全因素剧增。

另外，2001 年中国加入世界贸易组织（WTO）以来，中美、中欧之间的贸易摩擦，尤其是软件版权纠纷不断加剧。考虑到国家信息安全战略因素以及知识产权保护因素，为了大力发展、培养自己的软件企业，中国政府、高校越来越重视数据安全、版权制度建设以及知识产权的保护。学习、使用、开发、研究开放源代码的软件（简称开源软件），不仅可以有效缩短国内软件与国外软件之间的差距，还可以减少经费的投入。大家所熟知的 Android 操作系统、Linux 操作系统、MySQL 数据库管理系统、Apache 服务器软件都是开源软件，有些开源软件（例如 Android 操作系统）不仅改变了人们的生活，而且改变了企业的命运。开源软件的学习，使用、开发、研究势在必行。

与上述商业化的数据库管理系统相比，MySQL 具有开源、免费、体积小、便于安装，而且功能强大等特点。很多企业将 MySQL 作为首选数据库管理系统，MySQL 成为了全球最受欢迎的数据库管理系统之一。目前，淘宝、百度、新浪微博已经将部分业务数据迁移到 MySQL 数据库中，MySQL 的应用前景可观。

考虑到 MySQL 成本低廉、开源、免费、易于安装、性能高效、功能齐全等特点，因此 MySQL 非常适合教学。从 2013 年下半年开始，全国计算机等级考试二级中新增了“MySQL 数据程序设计”考试科目，足以看到教育部门对开源软件学习、使用、开发、研究的重视程度。目前越来越多的高校相继开设了 MySQL 课程，然而相对 SQL Sever、Oracle 等商业数据库管理系统而言，市场上适合中国读者学习的 MySQL 书籍少之又少，这也是本书编写的初衷，希望本书能够为各类读者提供一条学习 MySQL 的捷径，更希望本书能够帮助更多的读者走进数据库开发的殿堂。



本章主要讲解关系数据库设计的相关知识，本书从第 2 章开始，才会讲解 MySQL 的相关知识。

## 1.1.2 关系数据库

目前主流的数据库使用的“数学模型”是“关系”数据模型（简称关系模型），使用关系模型对数据进行组织、存储和管理的数据库称为关系数据库。关系数据库中所谓的“关系”，实质上是一张二维表。

以“选课系统”为例，教师申报课程相关信息（其中包括课程名、人数上限、任课教师及

课程描述等信息),并将课程信息录入到“选课系统”数据库的课程表(二维表)中,如表 1-1 所示。管理员从课程表(二维表)中获取课程信息,并对课程进行审核(修改课程表中课程的状态信息);接着学生可以从课程表(二维表)中获取已经审核的课程信息进行浏览,然后选修自己感兴趣的课程,填入选课表中(二维表);期末考试结束后,任课教师把学生的考试成绩录入到选课表(二维表)中……越来越多的二维表就构成了“选课系统”数据库。可以看出,一个数据库通常包含多个二维表(称为数据库表,或者简称为表),从而实现某个应用各类信息的存储和维护。

数据库表是由列和行构成的,表中的每一列(也叫字段)都由一个列名(也叫字段名)进行标记;除了字段名那一行,表中的每一行称为一条记录。表 1-1 所示的课程表共有 5 个字段以及 5 条记录。外观上,关系数据库中的一个数据库表和一个不存在“合并单元格”的电子表格(例如 Excel)相同。与电子表格不同的是:同一个数据库表的字段名不能重复。为了优化存储空间以及便于数据排序,数据库表的每一列必须指定某种数据类型。当然,数据库表与电子表格的区别并不局限于此,随着学习的深入,读者可以了解它们之间更多的区别。

表 1-1 课程表(二维表)

课程名	人数上限	任课教师	课程描述	状态
java 语言程序设计	60	张老师	暂无	未审核
MySQL 数据库	150	李老师	暂无	未审核
C 语言程序设计	60	王老师	暂无	未审核
英语	230	马老师	暂无	未审核
数学	230	田老师	暂无	未审核

需要注意的是,作为数据库中最为重要的数据库对象,数据库表的设计过程并非一蹴而就,表 1-1 所示的课程表根本无法满足“选课系统”的功能需求(甚至该表就是一个设计失败的数据库表)。事实上,数据库表的设计过程并非如此简单,本章的重点就是讨论如何设计结构良好的数据库表。数据库中还包含其他数据库对象,如触发器、存储过程、视图、索引、函数等,这些知识将在后续章节进行详细讲解。

### 1.1.3 结构化查询语言 SQL

结构化查询语言(Structured Query Language, SQL)是一种应用最为广泛的关系数据库语言,该语言定义了操作关系数据库的标准语法,几乎所有的关系数据库管理系统都支持 SQL。使用 SQL 可以轻松地创建、管理关系数据库的各种数据库对象以及维护数据库中的各种数据,例如删除“选课系统”中课程表(course)的所有记录,使用结构化查询语言“delete from course”语句可以轻松地实现。

SQL 仅仅提供了一套标准语法,为了实现更为强大的功能,各个关系数据库管理系统都对 SQL 标准进行了扩展,典型的有 Oracle 的 PL/SQL,SQL Server 的 T-SQL。MySQL 也对 SQL 标准进行了扩展(虽然至今没有命名),例如,MySQL 命令“show databases;”用于查询当前 MySQL 服务实例所有的数据库名,该命令是 MySQL 的特有命令,并不是 SQL 标准中定义的 SQL 语句,该命令在其他数据库管理系统中运行时将报错,例如,在 SQL Server 中运行该命令时,显示“未能找到存储过程 'show'”错误信息。这些扩展命令导致了各个数据库产品之间的差异,这种差异为同一个数据库在不同数据库产品之间的移植带来诸多不便。



说明

为了区分 SQL 扩展以及 SQL 标准, 本书将符合 SQL 标准的代码称为“SQL 语句”, 将 SQL 扩展部分的代码称为“MySQL 命令”或者“MySQL 语句”。例如“delete from course”是 SQL 语句; “show databases;”是 MySQL 命令或者 MySQL 语句。

SQL 并不是一种功能完善的程序设计语言, 例如, 不能使用 SQL 构建人性化的图形用户界面 (Graphical User Interface, GUI), 程序员需要借助 Java、VC++ 等面向对象程序设计语言或者 HTML 的 FORM 表单构建图形用户界面 (GUI)。如果选用 FORM 表单构建 GUI, 程序员还需要使用 JSP、PHP 或者 .NET 编写 Web 应用程序, 处理 FORM 表单中的数据以及数据库中的数据, 这些知识将在“网上选课系统的开发”章节中进行详细讲解。

## 1.2 数据库设计的相关知识

数据库设计是一个“系统工程”, 要求数据库开发人员:

- 熟悉“商业领域”的商业知识, 甚至是该商业领域的专家。
- 利用“管理学”的知识与其他开发人员进行有效沟通。
- 掌握一些数据库设计辅助工具。



说明

本书提到的数据库开发人员指的是能够从事各种应用系统的数据库开发工作的相关人员, 主要包括能够从事需求分析、数据库建模、数据库设计、数据库实施及编写函数、存储过程或者触发器等数据库开发工作的相关人员。限于篇幅, 本书将选择一个大小合适、认知度合适的案例展现数据库设计、开发的所有流程, 并对该案例的应用程序使用软件工程的思想进行开发。

### 1.2.1 商业知识和沟通技能

数据库中存储的数据是“商业领域”的信息, 使用数据库技术可以解决“商业领域”的“商业问题”。对于数据库开发人员而言, 商业知识和沟通技巧永远是避不开的话题。数据库开发人员必须熟悉某种商业领域的商业知识, 甚至是该商业领域的专家, 才能使用数据库技术解决商业问题。试想一个不熟悉、不了解金融服务业 (或者制造业、零售业等行业) 运作流程的数据库开发人员, 即便掌握了数据库开发的所有技能, 也不可能设计一个结构良好的金融服务业 (或者制造业、零售业等行业) 数据库。

设计数据库时, 数据库开发人员经常与其他开发人员 (包括最终用户) 一起工作, 并且需要使用“管理学”的知识与其他开发人员进行有效沟通, 获取所需商业信息, 从而解决商业问题。因此, 对于数据库开发人员而言, 沟通的技巧也不能小觑。

熟悉一种“商业领域”的商业知识需要花费大量的时间, 很多数据库开发人员用毕生精力研究某个特定行业, 从而成为该“商业领域”的专家, 继而可以成功地设计该“商业领域”的数据库。同样对于读者而言, 必须了解某一“商业领域”的商业知识, 才能将数据库技术应用到该“商业领域”, 解决该“商业领域”的“商业问题”, 进而才能更有效地学习数据库的相关知识。

鉴于多数读者有过“网上选课”的经历, 限于篇幅, 本书选用“选课系统”作为案例, 尽量避开“商业领域”和“管理学”相关知识的讲解, 着重讲解数据库设计、开发过程中使用到的各

种数据库技术。通过该案例的讲解，读者能够在最短的时间内具备一定的数据库设计、开发能力，继而能够尽快地掌握使用数据库技术解决“商业问题”的能力。

## 1.2.2 数据库设计辅助工具

数据库开发是软件开发过程中一个非常重要的环节，甚至是一个核心环节。软件开发过程中，软件开发人员经常使用一些辅助工具提高软件开发的速度与质量，典型的辅助工具包括模型、工具和技术。这些辅助工具由软件开发专业人员根据自身经验提炼而成，且日益成熟。开发数据库时，数据库开发人员同样也需要使用一些数据库设计的辅助工具，从而提高数据库的开发速度与质量。

### 1. 模型

软件开发时经常使用到一些模型，模型是现实世界中事物特征与事物行为的抽象。模型包括数学模型（例如数学公式）、描述模型（例如报表、列表、备忘录等）和图形模型（例如 E-R 图、数据流程图 DFD、类图等）。对事物的特征进行抽象的过程称为数据建模。软件开发过程中通过数据建模，可以得到软件系统的 E-R 图或者类图等数据模型。对事物的行为进行抽象的过程称为业务建模。软件开发过程中通过业务建模，可以得到软件系统的程序流程图、数据流程图 DFD、时序图、状态图等业务模型。

一般而言，数据库设计更侧重于数据建模，程序设计更侧重于业务建模。然而在真实的软件开发环境中，数据建模与业务建模两者相辅相成，不可或缺。E-R 图是关系数据库数据建模过程中经常使用的数据模型。本章将对 E-R 图进行详细讲解，其他模型的相关知识请读者参考软件工程类的书籍，限于篇幅，本书不再赘述。

### 2. 工具

软件开发时经常使用到一些工具，这些工具为创建模型或其他组件提供了软件支持。例如在系统规划阶段<sup>①</sup>，需求分析人员经常使用软件项目管理工具为任务分配资源、跟踪进度以及管理预算，常用的软件项目管理工具是美国微软公司的 Project。在系统分析与设计阶段，需求分析人员经常使用计算机辅助系统工程工具（Computer Aided Software Engineering, CASE）进行数据建模以及业务建模，常用的 CASE 工具有 ERwin、PowerDesigner、Rational Rose 以及 Visio 等。在系统实施阶段，编程人员经常使用集成开发环境（Integrated Development Environment, 简称 IDE 工具）进行软件编码、编译、调试等工作，常用的 IDE 工具有 VC++6.0、Visual Studio、Eclipse 以及 NetBeans 等。在测试阶段，测试人员经常使用测试工具进行单元测试、功能测试以及性能测试，常用测试工具有 Junit 单元测试工具、QuickTest Professional 功能测试工具以及 LoadRunner 性能测试工具等。

在关系数据库数据建模时，数据库开发人员经常使用 ERwin、PowerDesigner、Visio 等 CASE 工具创建 E-R 图，甚至使用 ERwin、PowerDesigner、Visio 等工具直接创建数据库（例如 MySQL 数据库）或者直接生成 SQL 脚本文件（例如 MySQL 的 SQL 脚本文件）。

### 3. 技术

软件开发时使用的技术是一组方法，常用的技术包括：面向对象分析和设计技术、结构化分析和设计技术、软件测试技术和关系数据库设计技术等。其中，关系数据库设计技术决定了关系数据库设计的质量，这也是本章着重讲解的内容。关系数据库设计技术包含 E-R 图绘制以及关系

<sup>①</sup> 软件的开发不是一蹴而就的，通常分为若干个开发阶段：系统规划、系统分析、系统设计、系统实施、运行维护阶段以及测试阶段，其中测试阶段应该贯穿其他几个阶段。这些开发阶段构成了软件开发生命周期（Systems Development Life Cycle, SDLC）。

数据库设计两方面的内容，这两方面的内容稍后进行详细讲解。



数据建模制作 E-R 图的过程中，本章使用的 CASE 工具是 PowerDesigner。部分读者可能没有使用过 PowerDesigner，但笔者认为软件开发（包括数据库开发）是一种高级脑力劳动，工具代替不了软件开发人员以及数据库开发人员的“智慧”及“思想”，掌握这些“智慧”、“思想”对于数据库开发人员至关重要，这也是本书着重阐述的内容。读者在学习本章内容时，可以使用笔、纸或者绘图工具（例如 Word 绘图）设计 E-R 图，掌握本章的知识后，有精力的读者可以学习一下 ERwin、PowerDesigner 或者 Visio 工具的使用。

### 1.2.3 “选课系统”概述

相信大多数读者有过网上选课的经历，熟悉“选课系统”的基本操作流程，多数读者可以称得上是“选课”领域的“专家”，这为设计一个结构良好的“选课系统”数据库奠定了坚实的基础。为了将“选课系统”案例融入到数据库设计以及 MySQL 的各个知识点，限于篇幅，本书在不影响“选课系统”核心功能的基础上，适当地对该系统进行“定制”、“扩展”以及“瘦身”，“选课系统”的操作流程如图 1-2 所示。“选课系统”操作流程的文字描述如下。本书后续所有章节的内容，全部围绕该“描述”设计，开发“选课系统”的数据库表、索引、视图、函数、存储过程、触发器等数据库对象。

- 游客用户只能浏览已经审核的课程信息，不能享受“选课系统”提供的其他服务。游客用户注册成为学生或者教师，成功登录“选课系统”后，才能享受“选课系统”提供的其他服务。

- 教师成功登录“选课系统”后，可以申报选修课程，要求选修课程面向全校学生。

- 为保证教学质量，每一位教师只能申报一门选修课程。由于很多课程需要在教室内完成教学，因此课程的人数上限受到教室座位数量的限制（共有 60 座位、150 座位和 230 座位 3 种教室）。教师申报选修课程时需提供课程的详细信息，其中包括课程名、工号、教师姓名、人数上限、教师联系方式以及课程详细描述等。

- 教师申报课程信息后，经管理员审核通过才能供学生选修。

- 学生成功登录“选课系统”后，才可以进行选课。学生选课时，每位学生可以浏览所有已审核的课程信息，并进行选修。为保证学习效果，限制每位学生最多选修两门课程。学生选课时需提供学号、姓名、班级名、所属院系名和联系方式等信息，由系统自动记录选择课程的时间。

- 选课结束前，学生可以退课、调课。

- 选课结束后，当某一门课程的选修人数少于 30 人时，为避免教师、教室资源浪费，管理

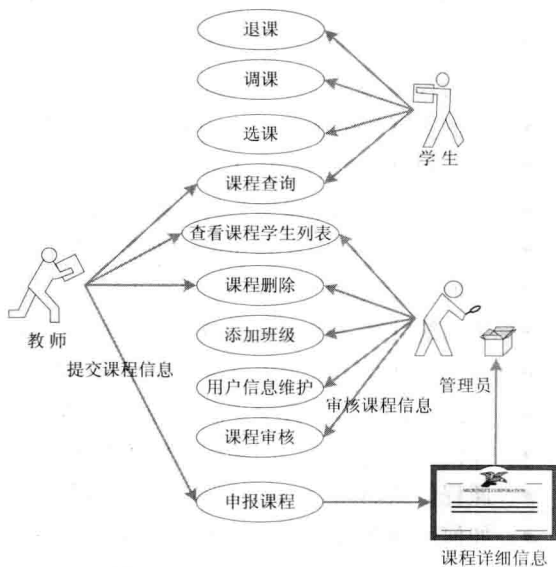


图 1-2 选课系统操作流程

教师申报选修课程时需提供课程的详细信息，其中包括课程名、工号、教师姓名、人数上限、教师联系方式以及课程详细描述等。



员有权删除该课程信息。某一门的课程信息删除后，选择该课程的学生需要重新选修其他课程。

- 管理员负责审核课程，添加班级信息（且班级名不能重复），以及维护用户信息。
- 教师可以查看本人申报课程的学生信息列表，管理员可以查看所有申报课程的学生信息列表。



说明

本书第 1~9 章介绍“选课系统”数据库开发流程，第 10 章介绍“选课系统”应用程序开发的相关知识，前 10 章内容循序渐进，章节之间知识衔接非常紧密，并且章节之间尽量避免知识重复和交叉，建议读者按照本书章节的顺序学习前 10 章的内容。

## 1.2.4 定义问题域

定义问题域是数据库设计过程中重要的活动，它的目标是准确定义要解决的商业问题。使用数据库技术可以解决“选课系统”存在的诸多“商业”问题，其中包括以下内容。

- 如何存储以及维护课程、学生、教师以及班级的详细信息？
- 不同教师申报的课程名能否相同？如果允许课程名相同，如何区分课程？
- 如何控制每位教师只能申报一门选修课程？
- 如何控制每门课程的人数上限在（60、150、230）中取值？
- 如何控制每一门课程的选课学生总人数不超过该课程的人数上限？
- 如何实现学生选课功能、退选功能以及调课功能？
- 如何控制每位学生最多可以选修两门课程，且两门课程不能相同？
- 系统如何自动记录学生的选课时间？
- 如何统计每一门课程还可以供多少学生选修？
- 如何统计人数已经报满的课程？
- 如何统计某一个教师已经申报了哪些课程？
- 如何统计某一个学生已经选修了多少门课程，是哪些课程？
- 如何统计选修人数少于 30 人的所有课程信息？
- 如何统计选修每一门课程的所有学生信息？
- 课程信息删除后，如何保证选择该课程的学生可以选修其他课程？
- 如何通过搜索关键字检索自己感兴趣的课程信息？

上述所有“商业”问题，都可以通过数据库技术找到答案，并可以在本书找到解决方案。有些“商业”问题可以使用数据库设计知识在本章进行解答；有些“商业”问题需借助具体的数据库管理系统（例如 MySQL）的知识，可以在其他章节中找到答案。

## 1.2.5 编码规范

结构化查询语言 SQL 是本书重点讲解的内容。一方面，数据库开发人员需要使用 SQL 编写部分业务逻辑代码（如触发器、存储过程、函数、事件等）完成部分业务功能。另一方面，程序开发人员需要在应用程序中构造 SQL 语句，实现应用程序与数据库的交互。为了保证数据库能够在不同的操作系统平台上进行移植，甚至为了保证应用程序能够在不同的数据库管理系统之间进行移植，数据库开发人员以及程序开发人员在书写 SQL 语句时需要遵循一些基本的编程原则，这些原则称为数据库编码规范。下面介绍一些常用的数据库编码规范，这些规范对任何一个追求高质量代码的人来