百战程序员丛书
新工科IT人才培养系列教材



MySQL 数据库教程

北京尚学堂科技有限公司 组编
刘凯立 张巧英 编著
高 淇 主审

西安电子科技大学出版社
http://www.xduph.com

需要全本请访问 www.ertongbook.com



百战程序员丛书

新工科 IT 人才培养系列教材

MySQL 数据库教程

北京尚学堂科技有限公司　组编

刘凯立　张巧英　编著

高　淇　主审



西安电子科技大学出版社

内 容 简 介

本书从初学者角度出发，通过精心设计的丰富示例由浅入深地讲解了 MySQL 数据库开发中的常用操作。全书共 18 章：第一至四章主要讲解了数据库的基础知识及操作，内容包括数据库概述、MySQL 的安装与配置、MySQL 支持的数据类型以及数据库的基本操作；第五至九章主要讲解了 MySQL 数据库的应用，内容包括表的基本操作，索引，插入、更新与删除数据，单表查询操作以及多表查询操作；第十至十二章主要讲解了 MySQL 数据库中常用的高级操作，内容包括事务、视图以及用户管理；第十三至十七章主要讲解了 MySQL 数据库中的存储过程、游标、存储函数、触发器、数据备份与恢复；第十八章主要讲解了关系型数据库设计的理论基础，内容包括数据模型、概念模型以及三大范式等。每章的最后都设计了针对该章内容的练习题，能够让读者趁热打铁，巩固所学的知识。

本书适合 MySQL 数据库的初学者快速入门，可作为广大高等院校计算机及其相关专业的教材，同时也可作为数据库开发人员的参考用书。

图书在版编目(CIP)数据

MySQL 数据库教程 / 刘凯立，张巧英编著. —西安：西安电子科技大学出版社，2019.6
ISBN 978-7-5606-5265-8

Ⅰ. ① M… Ⅱ. ① 刘… ② 张… Ⅲ. ① SQL 语言—程序设计—教材 Ⅳ. ① TP311.132.3

中国版本图书馆 CIP 数据核字(2019)第 037532 号

策划编辑　李惠萍　刘统军
责任编辑　王　斌　雷鸿俊
出版发行　西安电子科技大学出版社(西安市太白南路 2 号)
电　　话　(029)88242885　88201467　　邮　　编　710071
网　　址　www.xduph.com　　电子邮箱　xdupfxb001@163.com
经　　销　新华书店
印刷单位　陕西日报社
版　　次　2019 年 6 月第 1 版　2019 年 6 月第 1 次印刷
开　　本　787 毫米×1092 毫米　1/16　印　张　21.5
字　　数　508 千字
印　　数　1～3000 册
定　　价　52.00 元
ISBN 978-7-5606-5265-8/TP
XDUP 5567001-1

如有印装问题可调换

前　言

MySQL 是目前 IT 行业最流行的开放源代码的数据库管理系统，它是一款开源的、免费的、轻量级的关系型数据库，具有体积小、速度快、成本低、开放源码等优点，受到了业界人士的青睐，其发展前景无可限量。同时 MySQL 也是一个支持多线程、高并发、多用户的关系型数据库管理系统。市面上关于 MySQL 数据库学习的书籍非常多，但是有的过于深奥，让广大初学者望而生畏；有的过于浅显，浅入浅出，仅限于简单的示例；有的不注重实战，工作用不到的内容长篇大论，浪费读者时间。

北京尚学堂科技有限公司多年来一直从事高端 IT 教育，并且同国内外上千家企业有直接的用人合作，所以我们深知学员的需求和企业的技术要求。

学员需求：学习不枯燥，能够快速入门、实战操作，熟悉底层原理。

企业要求：程序员既有实战技能，可以快速上手，也内功扎实，熟悉底层原理，后劲十足。

因此，针对以上需求，北京尚学堂科技有限公司设计、推出了本书。

本书主要有以下几个特点：

1. 使用当前最主流版本的 MySQL 软件

本书中的 MySQL 软件的版本为 8.0.12，在此版本中新增或摒弃了旧版本中的一些操作，如果还是继续研读旧版本的读物，可能会给初学者在实际工作中带来困惑。在本书各章节知识点的讲解中，我们明确指出了 MySQL 8.0.12 与之前版本的不同，方便读者重点关注和进行区分。

2. 内容由易到难、由浅入深

本书从数据库的基础知识开始讲解，逐步深入到数据库的基本操作和应用，最后又讲解了数据库中常用的高级操作。这些内容由易到难、由浅入深，使学习循序渐进，适合初学者以及各高校学生阅读。

3. 注重实战应用

本书以示例驱动模式进行知识点的讲解，注重实战应用，以理论与实际相结合为原则，让读者边学边练，达到快速入门的目的，同时拥有更强的实战技能。本书在每一章中都精心设计了一个完整的案例，该案例涉及每一章中所有的知识点，这些知识点几乎涵盖了 MySQL 软件中常用的操作，旨在让 MySQL 数据库的初学者能够快速、深刻地理解这些知识点。

4. 注重原理的讲解

本书力求通过合适的示例和简明的语言给大家讲清楚原理，并将需要注意的问题和容易出现错误的地方重点标记，让读者不仅能够明确重点难点，也能对每个知识点做到“知其然也知其所以然”。

编　者

2018 年 11 月

需要全本请访问 www.ertongbook.com

北京尚学堂科技有限公司简介

北京尚学堂科技有限公司成立于2006年，旗下拥有北京尚学堂、百战程序员、优效学院三大子品牌。公司目前业务涵盖软件开发、技术培训、技术咨询、在线教育四大领域，事业部遍布北京、上海、广州、长沙、成都、太原、郑州、哈尔滨、深圳、武汉等十多个城市。

公司目前与北京大学软件工程国家研发中心联合研发“程序理解与代码正确性智能判断”技术；与奇虎360集团达成人才培育战略合作；连续多年被新浪网、腾讯网授予“中国好老师”、“金牌教育机构”等称号；2017年成为教育部首批协同育人合作企业，并与国内14所高校联合申报、获批14项校企协同育人项目。

2016年我公司组织研发部、培训部、技术咨询部等多部门50多位一线技术人员，推出“百战程序员”系列丛书，丛书涉及大数据、人工智能、Java语言、C语言、Python语言等领域。

公司官网：https://www.bjsxt.com。

丛书编写组邮箱：book@sxt.cn。

高淇老师邮箱：gaoqi@sxt.cn。

目　录

第一章　数据库概述 1
1.1　数据库简介 1
1.1.1　数据管理技术的发展过程 1
1.1.2　数据库系统相关概念 2
1.1.3　什么是 SQL 语言 4
1.1.4　如何访问数据库 5
1.2　MySQL 简介 6
1.2.1　数据库的分类 6
1.2.2　常见关系型数据库 7
1.2.3　MySQL 的优势 8
本章小结 9
练习题 9
第二章　MySQL 的安装与配置 10
2.1　下载 MySQL 10
2.1.1　在 Windows 平台下载 MySQL 11
2.1.2　在 Linux 平台下载 MySQL 13
2.2　在 Windows 平台下安装与配置 MySQL 13
2.2.1　MSI 版 MySQL 的安装与配置 13
2.2.2　ZIP 版 MySQL 的解压与配置 19
2.3　在 Linux 平台下安装与配置 MySQL 22
2.3.1　安装 MySQL 22
2.3.2　配置 MySQL 23
2.4　MySQL 的常用操作 25
2.4.1　启动与关闭 MySQL 服务 25
2.4.2　登录与退出 MySQL 数据库 27
2.4.3　修改 MySQL 配置 30
2.5　MySQL 常用图形化管理工具——Navicat 32
2.5.1　下载 Navicat 32
2.5.2　安装 Navicat 33
2.5.3　通过 Navicat 登录 MySQL 数据库 36
本章小结 37
练习题 37
第三章　MySQL 支持的数据类型 38
3.1　数值类型 39
3.1.1　整数类型 39
3.1.2　浮点数和定点数类型 41
3.1.3　BIT 类型 42
3.2　日期和时间类型 43
3.2.1　TIME 类型 44
3.2.2　DATE 类型 45
3.2.3　YEAR 类型 46
3.2.4　DATETIME 类型 46
3.2.5　TIMESTAMP 类型 47
3.3　字符串类型 48
3.3.1　CHAR 类型和 VARCHAR 类型 48
3.3.2　BINARY 类型和 VARBINARY 类型 49
3.3.3　TEXT 类型和 BLOB 类型 50
3.3.4　ENUM 类型 50
3.3.5　SET 类型 51
3.4　JSON 类型 51
本章小结 52
练习题 52
第四章　数据库的基本操作 53
4.1　创建数据库 53
4.1.1　使用 SQL 语句创建数据库 53
4.1.2　使用图形界面创建数据库 54
4.2　查看数据库 55
4.2.1　使用 SQL 语句查看数据库 55
4.2.2　使用图形界面查看数据库 56
4.3　修改数据库 57
4.3.1　使用 SQL 语句修改数据库 57
4.3.2　使用图形界面修改数据库 58
4.4　删除数据库 58

4.4.1 使用 SQL 语句删除数据库 59
4.4.2 使用图形界面删除数据库 59
本章小结 60
练习题 60
第五章 表的基本操作 61
5.1 创建表 61
5.1.1 使用 SQL 语句创建表 62
5.1.2 使用图形界面创建表 63
5.2 查看表 65
5.2.1 查看表的基本结构 65
5.2.2 查看表的详细结构 66
5.3 修改表 66
5.3.1 修改表名 67
5.3.2 修改字段的数据类型 68
5.3.3 修改字段名 70
5.3.4 增加字段 71
5.3.5 修改字段的排列位置 74
5.3.6 删除字段 75
5.4 删除表 76
5.4.1 使用 SQL 语句删除表 76
5.4.2 使用图形界面删除表 77
5.5 表的约束 78
5.5.1 完整性约束 78
5.5.2 主键约束 78
5.5.3 唯一约束 84
5.5.4 非空约束 89
5.5.5 默认值约束 92
5.5.6 自增约束 95
5.5.7 外键约束 98
本章小结 103
练习题 103
第六章 索引 104
6.1 索引简介 104
6.1.1 索引的概念 104
6.1.2 使用索引的原因 104
6.1.3 索引的分类 105
6.2 创建索引 106
6.2.1 在创建表时创建索引 108
6.2.2 为已存在的表创建索引 114
6.3 删除索引 121
6.3.1 使用“alter table”语句删除索引 122
6.3.2 使用“drop index”语句删除索引 122
6.4 使用图形界面操作索引 123
本章小结 125
练习题 125
第七章 插入、更新与删除数据 126
7.1 插入数据 126
7.1.1 为所有字段插入数据 126
7.1.2 为指定字段插入数据 128
7.1.3 使用“set”关键字为字段插入数据 129
7.1.4 同时插入多条数据 131
7.1.5 插入查询结果 133
7.2 更新数据 135
7.2.1 更新指定记录 135
7.2.2 更新全部记录 136
7.3 删除数据 137
7.3.1 删除指定记录 137
7.3.2 删除全部记录 138
7.3.3 使用“truncate”语句删除数据 139
7.4 使用图形界面操作数据 142
本章小结 143
练习题 143
第八章 单表查询操作 145
8.1 简单查询 147
8.1.1 所有字段的查询 147
8.1.2 指定字段的查询 149
8.1.3 去除重复记录的查询 150
8.1.4 使用算术运算符的查询 152
8.1.5 使用字段别名的查询 153
8.1.6 设置数据显示格式的查询 154
8.2 对查询结果排序 155
8.2.1 按照指定的单字段排序 155
8.2.2 按照指定的多字段排序 156
8.3 条件查询 158
8.3.1 使用比较运算符的查询 158

8.3.2 使用“[not] between...and...”的范围查询 160
8.3.3 使用“[not] in”的指定集合查询 161
8.3.4 使用“is [not] null”的空值查询 163
8.3.5 使用“[not] like”的模糊查询 164
8.3.6 使用“and”的多条件查询 167
8.3.7 使用“or”的多条件查询 168
8.4 限制查询 168
8.4.1 不指定初始位置的限制查询 169
8.4.2 指定初始位置的限制查询 169
8.5 函数查询 170
8.5.1 使用单行函数的查询 170
8.5.2 使用多行函数的查询 183
8.6 分组查询 186
8.6.1 使用“group by”的简单分组查询 186
8.6.2 使用“group by”与统计函数的分组查询 187
8.6.3 使用“group by”与“having”的分组查询 189
本章小结 190
练习题 190
第九章 多表查询操作 191
9.1 交叉连接查询 193
9.2 自然连接查询 195
9.3 内连接查询 196
9.3.1 等值连接查询 196
9.3.2 非等值连接查询 198
9.4 外连接查询 199
9.4.1 左外连接查询 200
9.4.2 右外连接查询 201
9.5 子查询 202
9.5.1 标量子查询 202
9.5.2 行子查询 203
9.5.3 列子查询 203
9.5.4 表子查询 207
9.5.5 使用子查询的注意事项 208
9.6 多表查询练习 209
本章小结 215
练习题 215
第十章 事务 217
10.1 事务简介 217
10.1.1 事务的概念 217
10.1.2 事务的特性 217
10.2 事务控制 218
10.2.1 事务的开启 219
10.2.2 事务的提交 222
10.2.3 事务的回滚 223
10.3 事务的隔离级别 224
10.3.1 “read uncommitted”隔离级别 224
10.3.2 “read committed”隔离级别 226
10.3.3 “repeatable read”隔离级别 229
10.3.4 “serializable”隔离级别 236
本章小结 239
练习题 239
第十一章 视图 240
11.1 视图简介 240
11.1.1 视图的概念 240
11.1.2 使用视图的原因 240
11.2 创建视图 241
11.2.1 在单表上创建视图 242
11.2.2 在多表上创建视图 244
11.2.3 在其他视图上创建视图 245
11.3 查看视图 246
11.3.1 使用“show tables”语句查看视图 246
11.3.2 使用“show table status”语句查看视图 246
11.3.3 使用“describe”语句查看视图 247
11.3.4 使用“show create view”语句查看视图 248
11.3.5 在 views 表中查看视图 249
11.4 修改视图 250
11.4.1 使用“create or replace”语句修改视图 250
11.4.2 使用“alter view”语句修改视图 251

11.5 删除视图 252
11.6 更新视图 253
11.6.1 使用“insert”语句更新视图 253
11.6.2 使用“delete”语句更新视图 254
11.6.3 使用“update”语句更新视图 256
11.6.4 更新视图时的限制条件 257
本章小结 262
练习题 262
第十二章 用户管理 263
12.1 权限表 263
12.1.1 mysql.user 表 263
12.1.2 mysql.db 表 265
12.1.3 其他权限表 266
12.2 用户管理 266
12.2.1 用户登录与退出 MySQL 数据库 267
12.2.2 创建普通用户 269
12.2.3 删除普通用户 271
12.2.4 修改密码 273
12.2.5 找回密码 278
12.3 权限管理 281
12.3.1 各种权限介绍 281
12.3.2 查看权限 283
12.3.3 授予权限 283
12.3.4 收回权限 285
本章小结 286
练习题 287
第十三章 存储过程 288
13.1 存储过程简介 288
13.1.1 存储过程的概念 288
13.1.2 存储过程的好处 288
13.2 使用存储过程 289
13.2.1 创建存储过程 289
13.2.2 执行存储过程 291
13.2.3 创建存储过程的要点 292
13.2.4 删除存储过程 304
本章小结 304
练习题 304
第十四章 游标 305
14.1 游标简介 305
14.2 使用游标 305
本章小结 308
练习题 308
第十五章 存储函数 309
15.1 定义存储函数 309
15.2 删除存储函数 311
本章小结 312
练习题 312
第十六章 触发器 313
16.1 创建触发器 313
16.2 删除触发器 314
16.3 使用触发器 314
16.3.1 insert 型触发器 314
16.3.2 update 型触发器 316
16.3.3 delete 型触发器 317
本章小结 319
练习题 320
第十七章 数据备份与恢复 321
17.1 需要备份数据的原因 321
17.2 数据的备份类型 321
17.3 备份数据 322
17.4 恢复数据 323
本章小结 325
练习题 325
第十八章 关系型数据库设计 326
18.1 设计关系型数据库 326
18.1.1 数据模型 326
18.1.2 概念模型 327
18.1.3 设计数据库的步骤 328
18.1.4 关系型数据库的设计原则 328
18.2 三大范式 328
18.2.1 第一范式(1NF) 329
18.2.2 第二范式(2NF) 329
18.2.3 第三范式(3NF) 331
本章小结 333
练习题 333
参考文献 334

第一章 数据库概述

在日常生活中，我们通常把食物放进冰箱存储起来；在学习中，学生的入学信息、考试成绩等档案资料都会存放到档案室；在工作中，公司员工的个人信息(包括职工编号、姓名、性别、籍贯等)会存放到表格中。我们可以形象地将上面提到的冰箱、档案室、表格看做是存放物品的仓库，我们可以根据需求随时在这些仓库中查找某一种食物或者查找某一个学生或员工的基本信息。

同样的道理，我们可以把数据库也看成是一个仓库，用它来存储和管理数据。在实际开发中，通常都是使用数据库来存储数据的。比如学校的学生管理系统就是使用数据库来存储每位学生的入学信息、考试信息、选课信息、教师信息等；再比如我们熟知的 12306 网上购票系统同样需要用数据库来存储用户信息、站点信息、车次信息、余票信息等。

数据库的应用已经无处不在，本章我们主要对数据库和常用的数据库管理系统 MySQL 的基本概念进行介绍。

1.1 数据库简介

1.1.1 数据管理技术的发展过程

数据管理是指对数据进行有效的分类、组织、编码、存储、检索和维护的过程，其目的是使数据能够充分且高效地发挥其作用。到目前为止，数据管理共历经了三个阶段：人工管理阶段、文件系统阶段、数据库系统阶段。其中数据库系统阶段是目前最高级的阶段，下面我们简单介绍这三个阶段的发展情况。

1. 人工管理阶段

自 1946 年 2 月第一台电子计算机诞生至 20 世纪 50 年代中期，计算机主要应用于科学计算。当时，计算机除了硬件设备外，并没有任何的软件可以用于存储数据，而使用的外存也只有磁带、卡片和纸带，并没有磁盘等直接存储设备；软件中只有汇编语言，没有操作系统。所以数据只能采用人工管理的方式。

人工管理阶段存在许多弊端，如下所述：

(1) 不能长期保存数据。由于数据存储在处理数据的程序中，导致数据与程序组成一个整体，程序运行时数据载入，程序结束时数据随着内存的释放而消失。即使是存储在磁带或卡片等外存中的数据，也只是一些临时数据。

(2) 没有软件对数据进行保存。程序设计者不仅要考虑数据之间的逻辑结构，还要考虑数据的存储结构、存取方式等。

(3) 数据面向应用(数据不能共享)。数据是附属于程序的，即使两个程序拥有相同的数据，也必须设计各自的数据存储结构和存取方式，还不能实现相同数据的共享，因此会导致程序与程序之间存在大量的重复数据。

(4) 数据不具备独立性。由于数据依托于程序，因此一旦数据的存储结构发生变化，就会导致程序的改变，使得数据没有独立性。

2. 文件系统阶段

20 世纪 50 年代后期到 60 年代中期，由于出现了磁盘、磁鼓等直接存储设备，软件也有了各种高级语言和操作系统，因此计算机不仅可以应用于科学计算，也被大量应用于经营管理活动。人们可以将程序所需的大量数据组织成数据文件，长期保存到直接存储设备中，然后利用操作系统中的文件管理功能随时对数据进行存取。

发展到文件系统阶段，对于数据的存储已经有了质的飞跃，该阶段的主要特点如下：

(1) 数据可以长期保存。数据保存在磁盘上，用户可以通过程序对数据进行增、删、改、查操作。

(2) 使用文件系统来管理数据。文件系统是程序与数据之间的接口，程序需要通过文件系统建立、存储和操作数据。

(3) 数据冗余大(数据共享性差)。因为文件是为特定的用途设计的，所以会造成数据在多个文件中被重复存储。

(4) 数据不一致。这是由于数据冗余和文件的独立性造成的，在更新数据时，很难保证相同数据在不同文件中的一致性。

(5) 数据独立性差。修改文件的存储结构后，相关的程序也需要修改。

3. 数据库系统阶段

20 世纪 60 年代后期，存储技术不断发展，出现了大容量的磁盘，因此计算机管理和处理的数据量急剧增加，原有的文件系统已经不能满足大量用户对数据共享性、独立性及安全性的需求，所以数据库应运而生。

1968 年，IBM 公司成功研发出数据库系统，这标志着数据管理技术进入了第三个阶段，即数据库系统阶段。在该阶段中，数据库替代了文件来存储数据，使得计算机能够更快速地处理大量的数据。数据库系统阶段弥补了文件系统阶段的不足，具有如下特点：

(1) 数据的结构化。通过存储路径实现记录之间的联系，这是文件系统所不具备的。

(2) 数据面向系统(数据实现了共享)。对于任何一个系统来说，数据库中的数据结构是透明的，任何程序都可以通过标准化接口来访问数据库。

(3) 数据的独立性强。数据的逻辑结构和物理结构实现了分离，用户以简单的逻辑结构操作数据即可，无需考虑数据的物理结构，转换工作由数据库管理系统实现。

(4) 数据的安全性。并非任意用户都可以存取数据库中的数据，数据库的安全性控制可以防止非法用户对数据的非法操作。

1.1.2 数据库系统相关概念

时至今日，我们仍然处于数据库系统阶段，因此我们必须要知道什么是数据，什么是数据库，什么是数据库管理系统，什么是数据库系统，等等。下面我们就来了解此部分内容。

需要全本请访问 www.ertongbook.com

1. 数据

数据(Data)是指对客观事物进行描述并可以鉴别的符号。这些符号是可识别的、抽象的。它不仅仅指狭义上的数字，而是有多种表现形式，如字母、文字、文本、图形、音频、视频等。现在计算机存储和处理的数据范围十分广泛，而描述这些数据的符号也变得越来越复杂。

2. 数据库

数据库(DataBase，DB)是指以一定格式存放、能够实现多个用户共享、与应用程序彼此独立的数据集合。

3. 数据库管理系统

数据库管理系统(DataBase Management System，DBMS)是指用来定义和管理数据的软件。如何科学地组织和存储数据，如何高效地获取和维护数据，如何保证数据的安全性和完整性，等等，这些都需要靠数据库管理系统来完成。目前比较流行的数据库管理系统有Oracle、MySQL、SQL Server、DB2 等。

4. 数据库应用系统

数据库应用系统(DataBase Application System，DBAS)是指在数据库管理系统的基础上，使用数据库管理系统的语法开发的直接面对最终用户的应用程序。例如，学生管理系统、人事管理系统、图书管理系统等。

5. 数据库管理员

数据库管理员(DataBase Administrator，DBA)是指对数据库管理系统进行操作的人员，其主要负责数据库的运营和维护。

6. 最终用户

最终用户(User)是指数据库应用程序的使用者。用户面向的是数据库应用程序(通过应用程序操作数据)，并不会直接与数据库打交道。

7. 数据库系统

数据库系统(DataBase System，DBS)一般是由数据库、数据库管理系统、数据库应用系统、数据库管理员和最终用户构成的。其中 DBMS 是数据库系统的基础和核心。图 1-1 所示为数据库系统组成示意图。

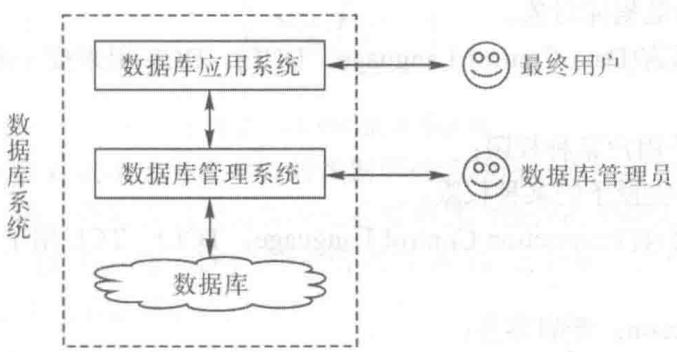


图 1-1　数据库系统组成

注意：对于初学者而言，特别容易混淆数据库与数据库系统的概念，以为数据库就是数据库系统，而实际上数据库系统的范围要比数据库大得多。

1.1.3 什么是 SQL 语言

我们都知道，数据库管理员(DBA)通过数据库管理系统(DBMS)可以对数据库(DB)中的数据进行操作，但具体是如何操作的呢？这就涉及我们本节要讲的 SQL 语言。

SQL(Structured Query Language)是结构化查询语言的简称，它是一种数据库查询和程序设计语言，同时也是目前使用最广泛的关系型数据库操作语言。在数据库管理系统中，使用 SQL 语言来实现数据的存取、查询、更新等功能。SQL 是一种非过程化语言，只需要提出“做什么”，而不需要指明“怎么做”。

SQL 是由 IBM 公司在 1974 至 1979 年之间以 E.J.Codd 发表的关系型数据库理论为基础开发的，其前身是“SEQUEL”，后更名为 SQL。由于 SQL 语言具有集数据查询、数据操纵、数据定义和数据控制功能于一体，类似自然语言、简单易用以及非过程化等特点，得到了快速的发展，并于 1986 年 10 月被美国国家标准学会(American National Standards Institute，ANSI)确定为关系型数据库管理系统的标准语言，后为国际标准化组织(International Organization for Standardization，ISO)定为国际标准。

SQL 语言分为以下几个部分：

(1) 数据查询语言(Data Query Language，DQL)。DQL 主要用于数据的查询，其基本结构是使用 select、from 和 where 子句的组合来查询一条或多条数据。

(2) 数据操作语言(Data Manipulation Language，DML)。DML 主要用于对数据库中的数据进行增加、修改和删除操作，主要包括：

① insert：增加数据；

② update：修改数据；

③ delete：删除数据。

(3) 数据定义语言(Data Definition Language，DDL)。DDL 主要针对数据库对象(表、索引、视图、触发器、存储过程、函数、表空间等)进行创建、修改和删除操作，主要包括：

① create：创建数据库对象；

② alert：修改数据库对象；

③ drop：删除数据库对象。

(4) 数据控制语言(Data Control Language，DCL)。DCL 用来授予或回收访问数据库的权限，主要包括：

① grant：授予用户某种权限；

② revoke：回收授予的某种权限。

(5) 事务控制语言(Transaction Control Language，TCL)。TCL 用于数据库的事务管理，主要包括：

① start transaction：开启事务；

② commit：提交事务；

③ rollback：回滚事务；

④ set transaction：设置事务的属性。

1.1.4　如何访问数据库

通过上述讲解可知，数据库管理系统可以使用SQL语言来操作数据库中的数据。其实，应用程序中也可以嵌套使用SQL语言来实现对数据库中数据的操作，但是如何才能让程序中的SQL语言发挥作用，这就涉及我们本节要讲解的数据库访问技术。

数据库技术经过多年发展，已出现了多种数据库访问技术，如ODBC(Open DataBase Connectivity，开放数据库互联)、DAO(Data Access Object，数据访问对象)、RDO(Remote Data Object，远程数据对象)、OLE DB(Object Linking and Embedding DataBase，对象链接和嵌入数据库)、ADO(Active Data Object，活动数据对象)、JDBC(Java DataBase Connectivity，Java数据库连接)等。随着时代的发展，DAO和RDO技术正在逐渐退出历史舞台。

注意：在使用数据库访问技术时，必须安装相应的驱动程序，不同的访问技术有不同的驱动程序。

1. ODBC

在数据库发展初期，不同类型的数据库之间很难实现数据交换。为解决这个问题，Microsoft开发了ODBC技术，它为编写关系型数据库的应用程序提供了统一的接口，使开发人员无需了解连接具体细节就可以访问数据库。

目前，大部分的关系型数据库都提供了ODBC驱动程序。ODBC的体系结构如图1-2所示。

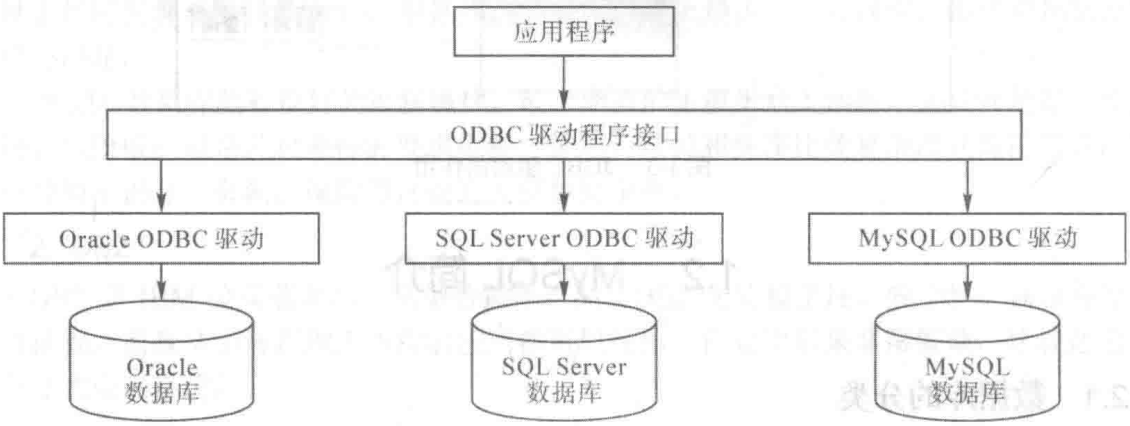


图1-2　ODBC的体系结构

我们可以将图1-2看成是应用程序访问数据库的流程图。应用程序直接和ODBC驱动程序管理器打交道，驱动管理器利用为各种关系型数据库提供的ODBC驱动程序来访问不同的数据库。ODBC驱动程序由ODBC驱动管理器调用，它会根据客户的请求调用相应的ODBC驱动，然后连接到数据库。

2. OLE DB

OLE DB是对ODBC的拓展，前者在后者基础上提供了COM接口，让应用程序能够

以统一的方式存取各种不同的数据源。实际开发中，数据可能存储在 Excel、E-mail 或者非关系型数据库中，而并非传统的关系型数据库，但是 ODBC 只能访问关系型数据库，所以 OLE DB 技术便应运而生。

3. ADO

我们可以将 ADO 看成是对 OLE DB 的封装。虽然 OLE DB 允许程序员访问各种类型的数据源，但是其非常底层化，编程非常困难，对程序员的水平有很高的要求。为了解决这个问题，Microsoft 推出了 ADO 技术，大大简化了程序员的工作，因此 ADO 越来越被程序员所喜爱。

4. JDBC

JDBC 是专门针对 Java 语言的一种数据库访问技术，是一种用于执行 SQL 语句的 Java API，为多种关系型数据库提供统一的访问接口。它是连接 Java 应用程序与数据库的桥梁。JDBC 由 Java 语言编写的类和接口组成。通过 JDBC 来操作数据库时，必须有相应的 JDBC 驱动。图 1-3 所示为 JDBC 驱动作用示意图。

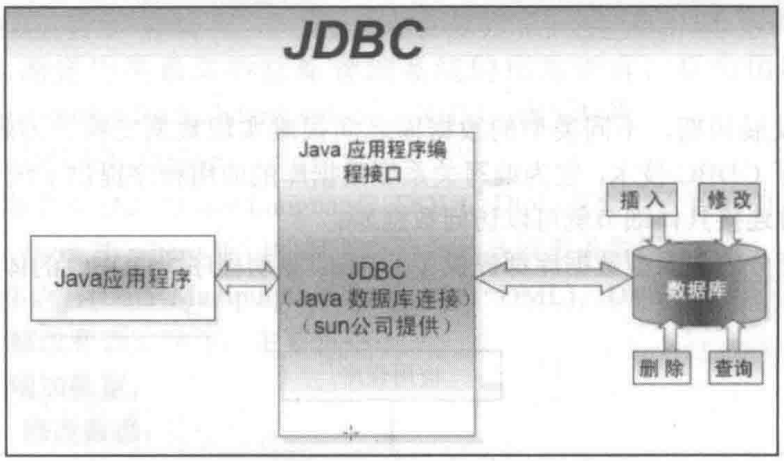


图 1-3　JDBC 驱动的作用

1.2　MySQL 简介

1.2.1　数据库的分类

数据库经过几十年的发展，出现了多种类型。根据数据的组织结构不同，主要分为网状数据库、层次数据库、关系型数据库和非关系型数据库四种。目前最常见的数据库类型主要是关系型数据库和非关系型数据库。

1. 关系型数据库

关系型数据库模型是将复杂的数据结构用较为简单的二元关系(二维表)来表示，如图 1-4 所示。在该类型数据库中，对数据的操作基本上都建立在一个或多个表格上，我们可以采用结构化查询语言(SQL)对数据库进行操作。关系型数据库是目前主流的数据库技术，

其中具有代表性的数据库管理系统有 Oracle、DB2、SQL Server、MySQL 等。

学号	姓名	性别	年龄
S070476	张三	男	24
S070477	李四	女	23
S070478	王五	男	24
S070479	赵六	女	25

编号	学号	课程	成绩
1	S070476	英语	78
2	S070477	Java	89
3	S070478	英语	83
4	S070479	JSP	96

图 1-4　关系型数据库的存储方式

2. 非关系型数据库(NOSQL)

NOSQL(Not Only SQL)泛指非关系型数据库。关系型数据库在超大规模和高并发的 Web 2.0 纯动态网站中已经显得力不从心，暴露了很多难以克服的问题。非关系型数据库的产生就是为了解决大规模数据集合多重数据种类带来的挑战，尤其是大数据应用难题。常见的非关系型数据库管理系统有 Memcached、MongoDB、Redis 等。

1.2.2　常见关系型数据库

虽然非关系型数据库的优点很多，但是由于其并不提供 SQL 支持、学习和使用成本较高并且无事务处理功能，所以本书的重点是关系型数据库。下面我们将简要介绍常用的关系型数据库管理系统。

1. Oracle

Oracle 数据库是由美国甲骨文公司开发的世界上第一款支持 SQL 语言的关系型数据库。经过多年的完善与发展，Oracle 数据库已经成为世界上最流行的数据库，也是甲骨文公司的核心产品。

Oracle 数据库具有很好的可移植性，能在所有的主流平台上运行，并且性能好、安全性高、风险低；但是其对硬件的要求很高，管理、维护和操作比较复杂而且价格昂贵，所以一般用于银行、金融、保险等行业的大型数据库中。

2. DB2

DB2 是 IBM 公司著名的关系型数据库产品。DB2 无论稳定性、安全性、恢复性等都无可挑剔，而且从小规模到大规模的应用都可以使用，但是用起来非常繁琐，比较适合大型分布式应用系统。

3. SQL Server

SQL Server 是由 Microsoft 开发和推广的关系型数据库，SQL Server 的功能比较全面、效率高，可以作为中型企业或单位的数据库平台。SQL Server 可以与 Windows 操作系统紧密结合，无论是应用程序开发速度还是系统事务处理运行速度，都能得到大幅度提升。但是，SQL Server 只能在 Windows 系统下运行。

4. MySQL

MySQL 是一种开放源代码的轻量级关系型数据库，MySQL 数据库使用最常用的结构

化查询语言(SQL)对数据库进行管理。由于 MySQL 是开放源代码的，因此任何人都可以在 General Public License 的许可下下载并根据个人需要对其缺陷进行修改。

由于 MySQL 数据库体积小、速度快、成本低、开放源代码等优点，现已被广泛应用于互联网上的中小型网站中，并且大型网站也开始使用 MySQL 数据库，如网易、新浪等。

1.2.3　MySQL 的优势

MySQL 数据库最初是由 MySQL AB 公司开发，几经辗转，最后成为 Oracle 公司的产品。MySQL 是目前 IT 行业最流行的开放源代码的数据库管理系统，同时它也是一个支持多线程、高并发、多用户的关系型数据库管理系统。MySQL 之所以受到业界人士的青睐，主要是因为其具有以下几方面的优点。

1. 开放源代码

MySQL 最强大的优势之一在于它是一个开放源代码的数据库管理系统。开源的特点是给予了用户根据自己的需求修改 DBMS 的自由。MySQL 采用了 General Public License，这意味着授予用户阅读、修改和优化源代码的权利，这样即使是免费版的 MySQL，其功能也足够强大，这也是为什么 MySQL 越来越受欢迎的主要原因。

2. 可移植

MySQL 可以在不同的操作系统下运行，简单地说，MySQL 可以支持 Windows 系统、UNIX 系统、Linux 系统等多种操作系统平台。这意味着在一个操作系统中实现的应用程序可以很方便地移植到其他的操作系统下。

3. 轻量级

MySQL 的核心程序完全采用多线程编程，这些线程都是轻量级的进程，它在灵活地为用户提供服务的同时，又不会占用过多的系统资源。因此，MySQL 能够更快速、高效地处理数据。

4. 成本低

MySQL 分为社区版和企业版，社区版是完全免费的，而企业版是收费的。即使在开发中需要用到一些付费的附加功能，MySQL 价格相对于昂贵的 Oracle、DB2 等也具有很大的优势。免费的社区版也支持多种数据类型和标准的 SQL 查询语言，能够对数据进行查询、增加、删除、修改等操作，所以一般情况下社区版就可以满足开发需求，而对数据库可靠性要求比较高的企业则可以选择企业版。

> **注意**：MySQL 社区版与企业版的主要区别是：
>
> ● 社区版包含所有 MySQL 的最新功能，而企业版只包含稳定之后的功能。换句话说，社区版可以理解为是企业版的测试版。
>
> ● MySQL 官方的支持服务只针对企业版，如果用户在使用社区版时出现了问题，MySQL 官方是不负责任的。

需要全本请访问 www.ertongbook.com