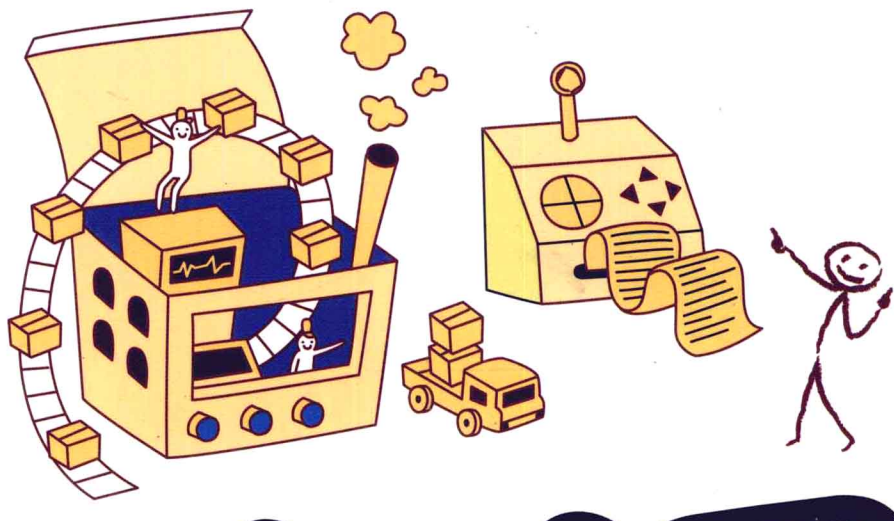


“这不是一本书，
而是一套通过研习数据库体系教会你快乐学习的方法”
从从根本上改变数据库学习方法，让编程趣味、互动起来.....

程序员，给哥来一个



颠覆你的
传统
学习方法



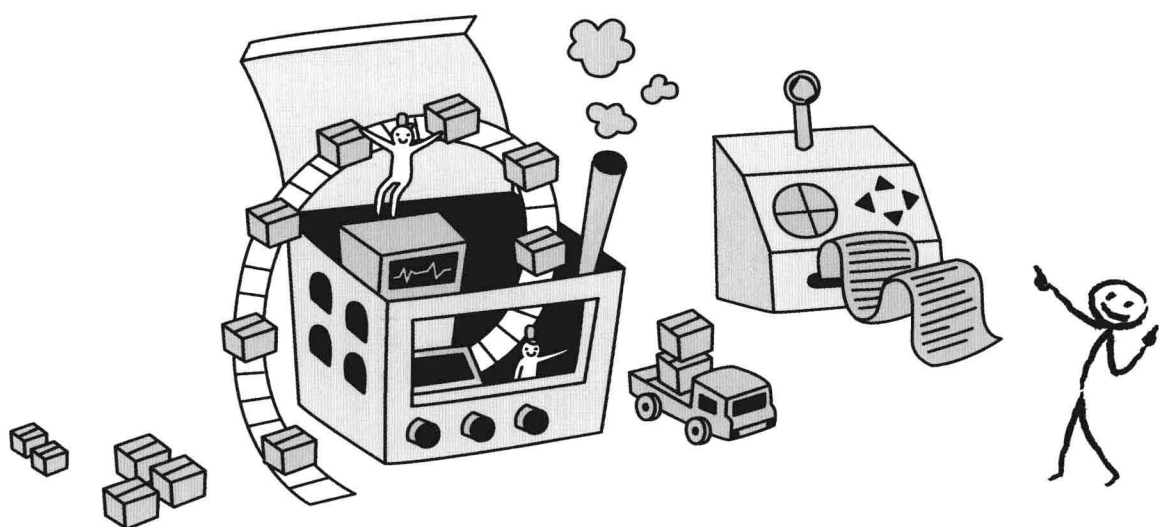
大话

数据库

邹茂扬 田洪川 编著



清华大学出版社



大话

数据库

邹茂扬 田洪川 编著

清华大学出版社
北京

内 容 简 介

本书是一本独特的数据库入门书，以最有效的教学思路讲解数据库的每一个知识点，完全以初学者的思维方式提出疑问再深入答疑。这也许不是一本传统的教科书，但绝对是自学数据库的首选书籍。本书采用【老田、小天】二人对话的形式讲解，其中不乏诙谐幽默的问题和解答，避免对知识点生搬硬套。

通过学习本书，你能够在嬉笑怒骂的环境中轻松掌握数据存储原理、数据库设计技巧以及大量数据库编程的实战经验，更重要的是能够掌握一种优秀的学习方法、解决问题的思路和思考的方式。这些经验和技巧得益于我和邹老师两人加起来近 25 年的项目开发和教育培训经历。

本书第一部分对于数据库的创建、备份、配置、安全等做详细介绍，通过这部分学习，可以掌握关系数据库的基础，以及对数据库的日常维护操作；本书第二部分对于分析项目需求，创建表，然后使用 T-SQL 语句和存储过程对表中数据做各种操作等做详细讲解，通过这一部分的学习，可以掌握对数据库的基本应用，熟练使用 T-SQL 语言建库、建表、T-SQL 查询、高级检索、存储过程、性能优化技巧等；

读者对象：希望靠一本书从头到尾自学的零基础学员；培训讲师的备课资料，因为这本书总结了我们的培训过程所遇到的问题和学生会问的问题、有疑虑的地方；自觉性不高的学员。

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。侵权举报电话：010-62782989 13701121933

图书在版编目(CIP)数据

大话数据库 / 邹茂扬，田洪川编著. —北京：清华大学出版社，2013.3

ISBN 978-7-302-30571-2

I. ①大… II. ①邹… ②田… III. ① 数据库系统—基本知识 IV. ①TP311.13

中国版本图书馆 CIP 数据核字(2012)第 261412 号

责任编辑：栾大成

装帧设计：杨玉芳

责任校对：徐俊伟

责任印制：李红英

出版发行：清华大学出版社

网 址：<http://www.tup.com.cn>, <http://www.wqbook.com>

地 址：北京清华大学学研大厦 A 座 邮 编：100084

社 总 机：010-62770175 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 刷 者：北京富博印刷有限公司

装 订 者：北京市密云县京文制本装订厂

经 销：全国新华书店

开 本：188mm×260mm 印 张：29.75 插 页：1 字 数：634 千字

版 次：2013 年 3 月第 1 版

印 次：2013 年 3 月第 1 次印刷

印 数：1~5000

定 价：59.00 元

前言

只有枯燥的教材，没有枯燥的编程

——天轰穿

“编程很枯燥！”是这句话是我要写一套编程类入门书籍的主要诱因。

而数据库则是作为进入编程世界的一个重要“关卡”，为什么这么说，首先，因为所有编程语言到了实际应用阶段都无法避免地与数据库打交道；其次，数据库是体现编程艺术的一个重要平台；另外，学好数据库，会使你以后学习其他语言的时候事半功倍。

而能否让你学得轻松、学得扎实就成为了一个至关重要的条件，这也是本书的宗旨。

通过学习本书，你能够在嬉笑怒骂的环境中轻松掌握数据存储原理、数据库设计技巧以及大量数据库编程的实战经验，更重要的是能够掌握一种优秀的学习方法、解决问题的思路和思考的方式。这些经验和技巧得益于我和邹老师两人加起来近 25 年的项目开发和教育培训经历。

本书特色

这不是一本以传统顺序堆砌而成的书

1. 本书以最有效的教学思路讲解数据库的每一个知识点，完全以初学者的思维方式提出疑问再深入答疑。这也许不是一本传统的教科书，但绝对是自学数据库的首选书籍。

2. 本书并非严格将数据库知识分类整理讲解，而是按照初学者的思维习惯，将每一个知识点放在最恰当的位置，所以单看目录，会感觉知识的排列不像同类书那样“井井有条”。

3. 本书采用【老田、小天】二人对话的形式讲解，其中不乏诙谐幽默的问题和解答，避免对知识点生搬硬套。

4. 本书总是提出问题再来解释，通过解释的过程来讲解新的知识。这样极大地避免了知识点的生硬出现，转而将学习的过程变成了解决问题的过程，同时也复习的相关其他知识点。

5. 本书中出现的专业术语随着知识的深入而出现，故尽量从头开始阅读。

6. 每章最后的“每日一练”中提出的问题常常有错误的问法夹杂其中，在这种题下面会紧跟着一道题，要求你将前一个题修改正确，极大避免了填鸭式教学，让你想不思考都不行。

7. 本书配备对应的视频教程，去百度谷歌“天轰穿趣味编程”就可以找到，或者去学云网搜“天轰穿”。

这是一本教会你学习方法的书

- 学习方式是按照初学者的理解方式，看实例→提出问题→解答问题；
- 通过对小天提出问题的解答来引导学员的思考和学习；
- 学习时间按“天”计算；
- 每章均有本章学习线路提示。

读者定位

- 希望靠一本书从头到尾自学的零基础学员；
- 培训讲师的备课资料，因为这本书总结了我们的培训过程所遇到的问题和学生会问的问题、有疑虑的地方；
- 自觉性不高的学员。

关于本书的创作起点与过程

我在 2006 年制作了《天轰穿 VS 2005 入门.Net2.0 系列视频教程》，其“非主流”的讲授风格受到大部分兄弟姐妹的肯定。截至目前，该视频在 6 年时间，已知的浏览量超过 1000 万次。

从 2007 年开始做培训，直到现在，我带过完全零基础的社招培训班、去高校上过专业课、去企业做过专题培训，也做了大量的以“天轰穿”命名的视频教程。在积累了大量实体培训和与网络学员交互培训的经验后，我再次萌生了要写一套专门给自学的兄弟姐妹的教材，于是和成都信息工程学院邹茂杨老师联合编写了这本书。

我不想太多去谈这本书怎么样，但当你翻开本书，那些无伤大雅的小幽默和深入浅出的实例引导会让您觉得选择这本“由初中生+高校教授的诡异组合”撰写的教材来学习数据库是对的。因为本书不仅是我个人自学技巧和教学经验的深度体现，还是邹茂杨

老师十余年教学经验的总结和汇聚。

近几年，常常有去一些企业和高校做讲座的机会，总有学员问：“川哥，我英语不好，能学好编程吗？我数学不好，能学好编程吗”。我的回答永远都是：“只要你努力，只要你坚持，就肯定能学好编程”。

- 因为你底子再差，不会比我这个初中生更差；
- 因为你英语再差，不会比我这个初中英语最高成绩就没不及格过的家伙差；
- 因为你数学再差，不会比我这个因为不会计算圆柱体面积而被老师骂的刺头更差。

我从 1999 年第一次接触计算机，从连鼠标都不会玩的土包子到做出自己的网站用了不到一年（一个纯静态页面组成的图片网站），再从只会做 HTML 页面到做出第一个 ASP 的留言本用了一年，之后多次闭关学习新技术（最狠的一次为了管住自己的双脚，把眉毛剃掉）。

回想写这本书的过程，眼眶湿了。虽然今天已经是学云网 CEO，但作为一个 1996 年初中毕业就混社会的农村小伙子而言，这一路走来，有欢笑也有泪水，但更多是汗水。由此得出一个结论，要学好编程，不在于你智商多高，而取决于你能否坚持，取决于你是否勤奋。编程不是看书、看视频就能学会了，而是靠大量的练习——不断举一反三的练习。

出社会后这十几年，我养成了一个习惯，无论做什么事都会全力以赴（如果做不到，就人为斩断自己的一切退路），写书的时候也一样，我辞去公司的讲师工作、推掉所有找上门的外包项目和一些高校的课程安排，期间仅靠老婆的工资和我去企业做培训的收入来糊口，过程中也有两周写不完一章的情况，由于想不出更好的办法来将深奥的知识讲得足够有趣，很多时候觉得自己很笨、很失败，很想放弃。但在老婆和朋友们的鼓励下，我坚持了下来。在此，感谢我亲爱的老婆，感谢这一路走来所有支持我、理解我、鼓励我的兄弟姐妹们！谢谢你们！！

目 录

第一部分 关系型数据库基础

第 1 章 概述 3

- 1.1 什么是数据库 4
- 1.2 为什么要使用数据库 4
- 1.3 数据库的基本概念 6
- 1.4 数据库的历史 7
- 1.5 关系数据库 8
- 1.6 为什么选择 SQL Server 9
- 1.7 SQL 语言 10
- 1.8 SQL 标准 11
- 1.9 10 分钟探索 IDE 12
 - 1.9.1 启动和登录 SQL Server Management Studio 14
 - 1.9.2 修改登录验证模式 15
 - 1.9.3 注册数据库服务器 18
 - 1.9.4 修改数据库服务器属性 18
 - 1.9.5 启动和停止服务器 19
 - 1.9.6 创建查询 19
 - 1.9.7 使用指定数据库 19
- 本章小结 20
- 问题 20

第 2 章 创建和维护数据库 21

- 2.1 系统数据库 23
 - 2.1.1 修改系统数据 23
 - 2.1.2 查看系统数据 24
- 2.2 创建数据库 24
 - 2.2.1 使用 Transact-SQL 语句创建数据库 25
 - 2.2.2 查看数据库文件属性 27

2.3 数据库文件和文件组 28

- 2.3.1 数据库文件的类型 29
- 2.3.2 文件组 29
- 2.3.3 删除数据库文件 31
- 2.3.4 管理文件组 32
- 2.3.5 文件组的填充策略 33
- 2.3.6 优化数据库的策略 34
- 2.3.7 文件状态 35

2.4 数据库状态和选项 35

2.5 查看数据库 37

2.6 删除数据库 38

2.7 修改数据库 39

- 2.7.1 修改数据库名称 39
- 2.7.2 扩展数据库 40
- 2.7.3 收缩数据库 42

2.8 数据库快照 44

- 2.8.1 数据库快照的应用 45
- 2.8.2 数据库快照的原理 46
- 2.8.3 管理数据库快照 48

2.9 分离和附加数据库 51

- 2.9.1 分离数据库 51
- 2.9.2 附加数据库 52

2.10 移动数据库文件 54

2.11 移动和复制数据库 55

2.12 备份和还原数据库 57

- 2.12.1 备份数据库 58
- 2.12.2 还原数据库 60

本章小结 62

问题 63

第二部分 设计、实现和使用数据库

第3章 Transact-SQL 语言 67

3.1 SQL 与 Transact-SQL 语言

概述方式 69

3.2 Transact-SQL 语言的执行

方式与调试 71

3.2.1 调试代码 73

3.2.2 调试 Transact-SQL 代码 74

3.3 数据定义语言 (DDL) 75

3.4 数据操纵语言 (DML) 76

3.5 数据库控制语言 (DCL) 78

3.6 附加的语言元素 78

3.6.1 标识符和命名规范 79

3.6.2 局部变量 81

3.6.3 全局变量 82

3.6.4 运算符 83

3.6.5 表达式 88

3.6.6 注释 88

3.7 数据类型 90

3.7.1 字符数据类型 91

3.7.2 数字数据类型 93

3.7.3 日期和时间数据类型 95

3.7.4 二进制数据类型 97

3.7.5 其他数据类型 98

3.7.6 用户自定义数据类型 100

3.8 内置函数 103

3.8.1 概述 103

3.8.2 如何查看 SQL Server 帮助 中的语法 104

3.8.3 如何使用函数 109

3.8.4 函数类型 111

本章小结 133

问题 134

第4章 创建与维护表 135

4.1 概述 137

4.2 设计表时应该考虑的因素 137

4.3 E-R 模型 140

4.3.1 概述 141

4.3.2 属性和主键 142

4.3.3 外键 143

4.3.4 联系 145

4.3.5 关系规范化 147

4.4 利用 PowerDesigner 设计

数据库 150

4.4.1 PowerDesigner 说明和

模型设置 150

4.4.2 创建概念数据模型实体 153

4.4.3 创建概念数据模型关系 154

4.4.4 从概念数据模型到生成物理 数据模型 155

4.4.5 创建物理数据模型 157

4.4.6 更新已有的物理 数据模型 157

4.4.7 生成数据库脚本 158

4.5 表的基本特点和类型 159

4.5.1 表的特点 160

4.5.2 表的类型 160

4.6 创建和修改表 161

4.6.1 创建普通表 161

4.6.2 创建临时表 165

4.6.3 创建分区表 166

4.6.4 增加和删除列 171

4.6.5 修改列 173

4.6.6 创建和修改列标识符 174

4.6.7 查看表信息 178

4.6.8 删除表 179

4.7 约束 180

4.7.1 主键约束 181

4.7.2 外键约束 184

4.7.3 NOT NULL 约束 187

4.7.4	DEFAULT 约束	188	6.2.1	普通分组	231
4.7.5	CHECK 约束	190	6.2.2	使用 HAVING 子句	234
4.7.6	UNIQUE 约束	191	6.2.3	使用 ROLLUP 和 CUBE	234
4.7.7	禁止与删除约束	192	6.3	联合查询	237
本章小结		195	6.4	连接查询	240
问题		195	6.4.1	内连接	241
第 5 章 操作表中的数据		197	6.4.2	外连接	243
5.1	准备工作	199	6.4.3	交叉连接	246
5.2	插入语句	200	6.4.4	自连接	247
5.2.1	简单的插入语句	200	6.5	子查询技术	248
5.2.2	批量插入语句	204	6.5.1	使用 IN 和 NOT IN 的 子查询	248
5.3	检索数据	207	6.5.2	ANY、ALL 等比较运算符 的使用	249
5.3.1	选择数据列	207	6.5.3	使用 EXISTS 关键字	250
5.3.2	使用文字串	207	6.5.4	子查询的规则	251
5.3.3	改变列标题	208	本章小结		252
5.3.4	数据运算	209	问题		252
5.3.5	使用 ALL 与 DISTINCT 关键字	210	阶段作业		252
5.3.6	使用 TOP 关键字	212	第 7 章 索引		253
5.3.7	排序	213	7.1	概述	255
5.4	WHERE 子句	216	7.1.1	表组织	256
5.4.1	简单条件查询	216	7.1.2	堆	257
5.4.2	模糊查询	216	7.1.3	B 树	258
5.4.3	复合条件查询	218	7.2	索引的优缺点	258
5.4.4	使用 IN 子句	219	7.3	索引的类型	260
5.4.5	使用 BETWEEN 子句	220	7.3.1	聚集索引	260
5.4.6	空值与非空值	221	7.3.2	非聚集索引	261
5.5	修改语句	222	7.4	索引的属性	263
5.6	删除语句	223	7.4.1	唯一性索引	263
本章小结		224	7.4.2	复合索引	264
问题		224	7.5	创建索引	264
第 6 章 高级检索技术		225	7.5.1	使用向导创建索引	265
6.1	聚合技术	228	7.5.2	使用 CREATE INDEX 语句创建索引	267
6.1.1	SELECT 子句中的聚合	228	7.5.3	索引的选项	269
6.1.2	COMPUTE 子句中 的聚合	229	7.6	维护索引	270
6.2	分组数据	230			

7.6.1 查看索引碎片	270	9.2.3 GOTO 语句	305
7.6.2 重建索引	271	9.2.4 WHILE BREAK 和 CONTINUE 语句	306
7.6.3 统计信息	271	9.2.5 CASE 语句	309
7.6.4 查看索引	274	9.2.6 WAITFOR 语句	311
7.6.5 修改索引	274	9.3 游标	312
7.6.6 删除索引	276	9.3.1 游标的类型	313
本章小结	278	9.3.2 选择游标类型的原则	315
问题	280	9.3.3 游标的生命周期	315
第 8 章 视图	281	9.3.4 实现 Transact-SQL 游标	316
8.1 概述	283	9.4 用户自定义函数	327
8.1.1 为什么需要视图	283	9.4.1 创建用户自定义函数的 思考	330
8.1.2 什么是视图	283	9.4.2 用户自定义函数的分类	331
8.2 视图的优缺点	285	9.4.3 创建及使用用户自定义 函数	332
8.3 创建视图	286	9.4.4 维护用户自定义函数	338
8.3.1 创建视图的基本原则	287	本章小结	339
8.3.2 使用 SQL Server Management Studio 创建视图	288	问题	339
8.3.3 使用 Transact-SQL 命令创建视图	291	第 10 章 存储过程	341
8.4 使用视图	292	10.1 概述	343
8.5 查看视图	293	10.2 存储过程的优点	344
8.6 加密视图	294	10.3 存储过程的分类	345
8.7 修改视图	294	10.3.1 系统存储过程	345
8.8 删除视图	295	10.3.2 API 存储过程	346
8.9 重命名视图	295	10.3.3 用户自定义存储过程	347
8.10 通过视图更新数据	296	10.4 创建存储过程	348
8.10.1 通过视图插入数据	296	10.4.1 创建存储过程应考虑 的因素	348
8.10.2 使用 UPDATE 修改 数据	297	10.4.2 创建存储过程的语法	350
本章小结	298	10.4.3 创建不带参数的 存储过程	352
问题	298	10.4.4 创建带参数的存储过程	353
第 9 章 SQL 编程及高级应用	299	10.4.5 创建返回值的存储过程	358
9.1 概述	301	10.4.6 创建带有 OUTPUT 参数 的存储过程	361
9.2 流程控制语句	301	10.4.7 使用 SQL Server Management Studio	
9.2.1 IF...ELSE...语句	302		
9.2.2 BEGIN...END 语句	303		

创建存储过程	362	12.2.1 使用事务考虑的因素	403
10.5 维护存储过程	362	12.2.2 事务的声明和提交	404
10.5.1 查看存储过程信息	363	12.2.3 事务的回滚	407
10.5.2 加密存储过程	363	12.2.4 查看当前执行中的事务	409
10.5.3 修改、删除存储过程	364	12.2.5 事务的嵌套	410
10.6 存储过程进阶知识	364	12.3 事务的工作原理	412
10.6.1 存储过程的执行过程	365	12.4 锁定和行版本控制	413
10.6.2 存储过程命名	366	12.5 锁定的分类	415
本章小结	366	12.6 锁的自动优化	419
问题	367	12.6.1 升级阈值	420
第 11 章 触发器	369	12.7 死锁	422
11.1 概述	371	12.7.1 死锁的概念	422
11.1.1 触发器与约束规则	372	12.7.2 产生死锁的主要原因 和必要条件	423
11.1.2 触发器的优缺点	373	12.7.3 减少和预防死锁	424
11.2 触发器的分类	374	12.7.4 检测死锁	426
11.2.1 DDL 触发器	374	12.7.5 设置锁的优先级	430
11.2.2 登录触发器	375	本章小结	432
11.2.3 DML 触发器	377	问题	432
11.3 创建触发器	379	第 13 章 全文索引	433
11.3.1 创建 DDL 触发器	379	13.1 概述	435
11.3.2 可触发 DDL 触发器 的事件	382	13.2 全文索引概念	435
11.3.3 维护触发器	386	13.2.1 全文索引与查询	436
11.4 创建 DML 触发器	387	13.2.2 全文索引引擎	437
11.4.1 创建 DML 触发器 的语法	388	13.3 全文目录	438
11.4.2 创建 DML 触发器需要 考虑的因素	389	13.3.1 创建全文目录	438
11.4.3 创建 DML 触发器	391	13.3.2 修改全文目录	440
11.5 DML 触发器嵌套	394	13.3.3 查看全文目录	442
本章小结	397	13.4 管理全文索引	443
问题	397	13.4.1 创建全文索引需要考虑 的因素	443
第 12 章 事务和锁	399	13.4.2 创建全文索引	445
12.1 事务概述	401	13.4.3 查看全文索引	449
12.1.1 概念	401	13.4.4 修改和删除全文索引	450
12.1.2 属性	401	13.4.5 填充全文索引	451
12.2 创建事务	402	13.5 使用全文索引	452
		13.5.1 使用全文谓词 CONTAINS	

和 FREETEXT	加权词搜索	457
查询概述	13.5.7 使用 FREETEXT 查询 ..	458
13.5.2 使用 CONTAINS 谓词的	13.5.8 使用 CONTAINSTABLE	
简单搜索	函数搜索	459
13.5.3 使用 CONTAINS 谓词	13.5.9 使用 FREETEXTTABLE	
的派生词搜索	函数搜索	461
13.5.4 使用 CONTAINS 谓词	13.6 检索二进制列	461
的前缀词搜索	本章小结	462
13.5.5 使用 CONTAINS 谓词的	问题	463
邻近词搜索	结束语	464
13.5.6 使用 CONTAINS 谓词的		

第一部分 关系型数据库基础



第1章 概述

学习时间：第一天

地点：老田办公室

人物：老田、小天

本章要点

- 数据库是什么
- 为什么要使用数据库
- 数据库的基本概念
- 数据库的历史
- 数据库的分类
- SQL 语言及 SQL 标准
- 管理工具及其作用
- 针对熟悉工具的一系列小实践

本章学习线路

我们从一个问题来引入数据库是什么、为什么要用数据库、数据库的发展历史、SQL语言的概念和标准，最后将落点放到熟悉我们学习所用的工具使用实践。

全文以问答的方式，依照零基础学员的思维方式一步步地引导和讲解，以防止纯书面化那种严肃的文字风格导致学员无法耐心地去看完所有内容。

本书主要的出发点是引导一个完全没有编程思维的人一步步地具备这种以一个架构师的角度和方式去思考并解决问题的能力。为避免学员因为不熟悉工具而在学习中无谓地浪费时间，因此在本章后面有针对地对工具做了一系列的小实践。之后的章节则以开发和设计为主了。

1.1 什么是数据库

小天:老田,常听说学习编程必须学好数据库。这是为什么啊?数据库是干啥的呢?

老田:额……首先,数据库就是指存放数据的仓库,如同水库是存水的,军火库是放军火的。其次并非所有计算机程序都需要用到数据库,但是如果希望你的程序具备对大量数据的存储、整理、分析等,这就需要涉及数据库了。

小天:明白了,如果我的计算机程序只是存放极其少量的数据,则可以不必使用数据库,而是将数据存放在xml这类的文本文档中就OK了,只有当需要存储大量的数据,并且经常性地对这些数据进行操作时才需要用到数据库。但是我下一个问题又出来了,我问人家,我要学习编程,要学数据库的话学什么。人家好像说有什么SQL Server、MySQL、Oracle、DB2等,这些又是啥呢?不会是说学个编程要学这么多数据库吧?

老田:额……当然不用了,其实几种数据库除了各自的管理有些不同外,对于数据的操作都是大同小异的。因为SQL Server、MySQL、Oracle、DB2这么多数据库管理系统,我们完全可以简单地将它们理解为不同厂商或者组织开发的性质差不多的关系型数据库管理系统。对,它们就是一个数据库管理系统,进一步简化管理为存储和管理数据的工具就行了。

小天:不对啊,老田你在忽悠我吧?存储大量的数据和管理数据我觉得不一定要用数据库啊,我看很多人现在用Excel也挺不错的呀。

1.2 为什么要使用数据库

老田:我们来打个比方,如图1-1所示,你看看是不是类似这样的管理呢?

	A	B	C	D	E	F	G
1	员工姓名	目前所在区域	所属部门	部门主管	电话	所属分公司	公司领导
2	汪静远	北京	人力资源部	高大伟	15855445566	北京分公司	罗天伟
3	朱超	合肥	项目实施部	庾扯扯	15784547788	安徽分公司	黄世元
4	秦良骏	重庆	项目实施部	庾扯扯	36534324	重庆分公司	邹坤

图 1-1 Excel 表

小天:对啊,我觉得一目了然。

老田:那么当遇到频繁地做下面几种操作的时候你烦不烦:

- (1) 添加新员工;
- (2) 部门主管换电话了;
- (3) 部门换主管了;
- (4) 换公司领导了;
- (5) 多个员工换分公司;

(6) 某个员工换部门；

(7) 当员工数量超过500人。

接下来我们模拟做一下下面这几件事。

添加新员工：总体来说这个只要员工的归属资料齐全的话还是比较轻松的。万一不齐全就只好慢慢地一行行地查了。

部门主管换电话：如果你还不知道Excel可以全文搜索和替换的话呢，就只能一行行地改了，如果恰恰你之前录入某人资料的时候打错字了的话.....

小天：擦汗ing.....

老田：部门换主管，如果这个主管是直接走人了倒也干脆，否则的话先把新换上来的主管资料逐行改了还要继续为他安排新的工作。

当然，这些如果是数据量少其实都不是大问题了，但这个公司如果是中石油、中移动，那么数据量应该是不上万也是千吧？

还有个最严重的问题，你无法保证不会重复添加数据。

小天：别说啦，你说的这些情况我其实早烦透了，但是这些问题是否在你所谓的数据库中就解决了？如果真解决了，那快教我怎么用吧。

老田：当然解决了，因为我接下来要给你说的是关系型数据库，所以这些让人凌乱的关系问题都可以迎刃而解。比如上面这个难题，我们用数据库创建如下三张数据表，一次就解决了，以后再也不用担心那些麻烦事。

(1) 分公司表

分公司 ID	公 司 名	分公司领导	地 址
G1	北京总公司	罗天伟	北京 XX 路 XX 号
G2	安徽分公司	黄世元	合肥 XX 路 XX 号
G3	重庆分公司	邹坤	重庆 XX 路 XX 号

(2) 部门表

部门 ID	部门名称	部门领导	所属分公司的 ID
B1	人力资源部	高大伟	G1
B2	市场部	疯扯扯	G1
B3	综合管理部	李洁	G1
B4	项目实施部	天轰穿	G2

(3) 员工表

员工工号	姓 名	电 话	所属部门的 ID
U1	田皓文	18080801234	B2
U2	黄旺男	18080804321	B4

老田：发现上面三张表之间的关系没有？有什么好处？