高职高专 21 世纪规划教材

SQL Server 数据库原理及应用

曾长军　主编

曾长军　朱剑锋　刘坤　编著

人 民 邮 电 出 版 社

需要全本请访问 www.ertongbook.com

图书在版编目（CIP）数据

SQL Server 数据库原理及应用/曾长军主编；曾长军，朱剑锋，刘坤编著.
—北京：人民邮电出版社，2005.2
高职高专 21 世纪规划教材
ISBN 7-115-12829-4

Ⅰ. S… Ⅱ. ①曾…②朱…③刘… Ⅲ. 关系数据库—数据库管理系统，SQL Server—高等学校：技术学校—教材 Ⅳ. TP311.138

中国版本图书馆 CIP 数据核字（2005）第 006320 号

内 容 提 要

SQL Server 2000 是微软公司推出的 SQL Server 的新版本，是一个非常优秀的、大型的关系型数据库管理软件，方便使用，性能稳定，是电子商务等领域的最佳数据库产品之一。

本书从数据库的基本理论、基本知识出发，通过丰富的实例介绍数据库的分析、设计过程以及开发应用等。

本书主要介绍数据库的基本原理与基础知识，讲解 SQL Server 2000 的安装、使用、管理和维护等各个方面的知识，包括创建数据库、SQL 查询语句、T-SQL、索引与数据的完整性、存储过程和触发器、数据库的安全管理、备份和恢复等内容。最后两章从软件工程的观点讲解 Visual Basic/SQL Server 开发与编程，给出了一个 SQL Server 2000 应用综合实例。每章还配有一定数量的习题以帮助读者加深理解。

全书论述简明扼要，例证丰富，便于读者快速掌握 SQL Sever 2000 的基本操作与使用，可作为高职高专计算机专业“网络数据库”教材，也可作为其他专业的“数据库应用”、“电子商务数据库”或类似课程的教材，还可作为网站开发与维护人员学习用书。

高职高专 21 世纪规划教材
SQL Server 数据库原理及应用

◆ 主　　编　曾长军
◆ 编　　著　曾长军　朱剑锋　刘　坤
策划编辑　滑　玉
责任编辑　须春美

◆ 人民邮电出版社出版发行　　北京市崇文区夕照寺街 14 号
邮编　100061　　电子函件　315@ptpress.com.cn
网址　http://www.ptpress.com.cn
读者热线　010-67129259
北京隆昌伟业印刷有限公司印刷
新华书店总店北京发行所经销

◆ 开本：787×1092　1/16
印张：18.5
字数：438 千字　　2005 年 2 月第 1 版
印数：1 – 5 000 册　　2005 年 2 月北京第 1 次印刷

ISBN 7-115-12829-4/TP・4316

定价：24.00 元

本书如有印装质量问题，请与本社联系　电话：（010）67129223

目　　录

第 1 章　数据库基础概述 …… 1

1.1　数据库系统简介 …… 1
1.2　数据库系统的组成 …… 2
1.2.1　数据库 …… 2
1.2.2　数据库管理系统 …… 2
1.2.3　数据库系统的用户 …… 2
1.2.4　数据库系统的网络结构 …… 3
1.3　关系型数据库 …… 3
1.3.1　关系型数据库定义 …… 3
1.3.2　关系型数据库与表 …… 4
1.3.3　主键与外键 …… 4
1.3.4　字段约束 …… 5
1.3.5　数据完整性 …… 6
1.3.6　表的关联 …… 7
1.4　关系型数据库的设计 …… 8
1.4.1　E-R 模型 …… 8
1.4.2　数据库设计过程 …… 9
1.4.3　关系型数据库规范化分析 …… 10
习题 …… 13

第 2 章　SQL Server 2000 简介 …… 14

2.1　SQL Server 2000 的新特性 …… 14
2.1.1　数据库增强 …… 14
2.1.2　联合数据库服务器 …… 15
2.2　SQL Server 2000 的安装 …… 15
2.2.1　安装前的准备事项 …… 15
2.2.2　SQL Server 2000 的硬件和软件安装要求 …… 15
2.2.3　创建 SQL Server 2000 服务账号 …… 16
2.2.4　SQL Server 2000 的安装路径 …… 17
2.2.5　安装 SQL Server 2000 的步骤 …… 18
2.3　管理 SQL Server 2000 服务器组件 …… 23
2.3.1　SQL Server 2000 服务管理器 …… 23

2.3.2 SQL Server 2000 主要的服务器组件简介 24
2.4 SQL Server 2000 的通信组件 24
2.4.1 服务器端的网络实用工具 25
2.4.2 客户端网络实用工具 25
2.5 SQL Server 2000 主要的管理工具 26
2.5.1 企业管理器 26
2.5.2 查询分析器 27
2.6 注册服务器 27
2.7 一个简单的应用举例 27
习题 30

第 3 章 SQL Server 数据库管理 31

3.1 SQL Server 数据库简介 31
3.1.1 SQL Server 数据库 31
3.1.2 设计 SQL Server 数据库 31
3.1.3 数据库中的对象 32
3.1.4 SQL Server 数据库类型 32
3.1.5 文件和文件组 33
3.1.6 日志 33
3.2 创建数据库 34
3.2.1 使用 Transact-SQL 创建数据库 34
3.2.2 使用企业管理器创建数据库 36
3.3 修改数据库 38
3.3.1 使用 Transact-SQL 修改数据库 38
3.3.2 使用企业管理器修改数据库 39
3.4 删除数据库 40
3.4.1 使用 Transact-SQL 删除数据库 40
3.4.2 使用企业管理器删除数据库 41
3.5 查看与设置数据库参数 41
3.5.1 使用 sp_helpdb 查看数据库信息 41
3.5.2 使用企业管理器查看数据库信息 42
习题 43

第 4 章 表与视图 45

4.1 SQL Server 中的数据类型 45
4.1.1 数据类型 45
4.1.2 空值的含义 47
4.2 创建表 47
4.2.1 使用企业管理器创建表 47

4.2.2 使用 CREATE TABLE 语句创建表 48
4.3 修改表的结构 49
4.3.1 使用企业管理器修改表结构 49
4.3.2 使用 ALTER TABLE 语句修改表结构 49
4.4 添加表中的记录 50
4.4.1 使用企业管理器添加记录 50
4.4.2 使用 INSERT 语句添加记录 51
4.5 修改表中的记录 52
4.5.1 使用企业管理器修改记录 52
4.5.2 使用 update 语句修改记录 52
4.6 删除表中的记录 52
4.6.1 使用企业管理器删除记录 52
4.6.2 使用 DELETE 语句删除记录 53
4.7 删除表 53
4.7.1 使用企业管理器删除表 53
4.7.2 使用 DROP TABLE 语句删除表 54
4.8 视图的概念 54
4.9 创建视图 55
4.9.1 使用企业管理器创建视图 55
4.9.2 使用 CREATE VIEW 创建视图 57
4.10 管理视图 58
4.10.1 查看和修改视图信息 58
4.10.2 删除视图 59
4.11 通过视图修改基本表中的数据 59
4.11.1 查询操作 59
4.11.2 插入操作 60
4.11.3 修改操作 60
4.11.4 删除操作 61
习题 61

第 5 章 SQL（结构化查询语言） 62

5.1 SQL 62
5.1.1 SQL 概述 62
5.1.2 SQL 的分类 62
5.2 SELECT 查询语句 63
5.2.1 查询分析器 63
5.2.2 查询语句格式 64
5.3 基于单表的查询 65
5.3.1 查询表中指定的字段 65

需要全本请访问 www.ertongbook.com

5.3.2 通配符“*”的使用 …… 65
5.3.3 使用单引号加入字符串 …… 65
5.3.4 使用别名 …… 66
5.3.5 显示表达式的值 …… 67
5.3.6 使用 DISTINCT 短语消除重复的记录 …… 67
5.3.7 使用 WHERE 子句查询特定的记录 …… 67
5.3.8 使用 ORDER BY 子句对查询结果排序 …… 74
5.3.9 SQL 的聚合函数 …… 76
5.3.10 使用 GROUP BY 子句对查询结果进行分组 …… 79
5.3.11 使用 HAVING 子句筛选结果表 …… 80
5.4 基于多表的连接查询 …… 80
5.4.1 连接谓词 …… 81
5.4.2 以 JOIN 关键字指定的连接 …… 83
5.5 子查询 …… 88
5.5.1 使用 IN 或 NOT IN 的子查询 …… 89
5.5.2 INSERT、DELETE 和 UPDATE 语句中的子查询 …… 91
5.5.3 使用比较运算符的子查询 …… 93
5.5.4 使用 EXISTS 的子查询 …… 94
5.5.5 相关子查询 …… 96
5.5.6 使用 UNION 运算符组合多个结果 …… 96
5.5.7 在查询的基础上创建新表 …… 97
习题 …… 99

第 6 章 Transact-SQL …… 100

6.1 常量 …… 100
6.1.1 关于 SQL 标识符 …… 100
6.1.2 常量 …… 100
6.2 变量 …… 101
6.2.1 全局变量 …… 101
6.2.2 局部变量 …… 102
6.3 数据类型 …… 103
6.3.1 系统数据类型 …… 103
6.3.2 用户定义数据类型 …… 103
6.4 运算符与表达式 …… 105
6.5 流程控制语句 …… 107
6.5.1 BEGIN…END 语句 …… 107
6.5.2 IF…ELSE 语句 …… 107
6.5.3 无条件转移(GOTO)语句 …… 108
6.5.4 WHILE、BREAK 和 CONTINUE 语句 …… 108

6.5.5 RETURN 语句 …… 109
6.5.6 WAITFOR 语句 …… 109
6.6 系统内置函数 …… 110
6.6.1 系统函数 …… 110
6.6.2 日期函数 …… 112
6.6.3 字符串函数 …… 113
6.6.4 数据函数 …… 114
6.6.5 集合函数 …… 115
6.6.6 游标函数 …… 115
6.7 用户自定义函数 …… 115
6.7.1 用户函数的定义与调用 …… 115
6.7.2 用户定义函数的删除 …… 120
习题 …… 120

第 7 章 索引与数据完整性 …… 122

7.1 索引 …… 122
7.1.1 索引的分类 …… 122
7.1.2 索引的创建 …… 123
7.1.3 索引的删除 …… 127
7.2 数据完整性 …… 129
7.2.1 数据完整性的分类 …… 129
7.2.2 PRIMARY KEY 约束 …… 130
7.2.3 FOREIGN KEY 约束 …… 131
7.2.4 UNIQUE 约束 …… 132
7.2.5 CHECK 约束 …… 134
7.2.6 默认值约束 …… 135
习题 …… 137

第 8 章 存储过程和触发器 …… 139

8.1 存储过程概述 …… 139
8.1.1 存储过程的基本概念 …… 139
8.1.2 存储过程的类型 …… 139
8.2 存储过程的创建 …… 140
8.2.1 使用 CREATE PROCEDURE 语句创建 …… 140
8.2.2 使用企业管理器创建 …… 143
8.2.3 使用向导创建 …… 143
8.3 执行存储过程 …… 145
8.4 存储过程的参数 …… 147
8.4.1 参数传递方式 …… 147

8.4.2 使用默认参数 …… 149
8.4.3 使用带 OUTPUT 的返回参数 …… 149
8.5 存储过程的返回值 …… 150
8.5.1 RETURN 语句 …… 150
8.5.2 捕获存储过程的返回值 …… 151
8.6 存储过程的查看、修改和删除 …… 152
8.6.1 存储过程的查看 …… 152
8.6.2 存储过程的修改 …… 152
8.6.3 存储过程的删除 …… 153
8.7 触发器的概念 …… 153
8.8 触发器的创建 …… 154
8.8.1 使用企业管理器创建触发器 …… 154
8.8.2 使用 Transact-SQL 语句创建触发器 …… 154
8.9 触发器的使用 …… 157
8.9.1 inserted 表和 deleted 表 …… 157
8.9.2 INSERT 触发器和 UPDATE 触发器 …… 158
8.9.3 DELETE 触发器 …… 160
8.10 触发器的修改及删除 …… 161
8.10.1 触发器的修改 …… 161
8.10.2 触发器的删除 …… 162
习题 …… 162

第 9 章 备份恢复 …… 163

9.1 备份和恢复概述 …… 163
9.1.1 备份和恢复需求分析 …… 163
9.1.2 数据库备份的基本概念 …… 163
9.1.3 数据库恢复概念 …… 164
9.2 备份操作和备份命令 …… 164
9.2.1 创建备份设备 …… 164
9.2.2 备份命令 …… 165
9.2.3 使用企业管理器进行备份 …… 167
9.2.4 使用备份向导进行备份 …… 169
9.3 恢复操作和恢复命令 …… 172
9.3.1 检查点 …… 172
9.3.2 数据库的恢复命令 …… 173
9.3.3 使用企业管理器恢复数据库 …… 174
9.4 导入导出 …… 177
9.4.1 导入导出概念 …… 177
9.4.2 使用实用程序导入导出数据 …… 178

习题 179

第 10 章 系统安全管理 180

10.1 SQL Server 2000 的身份认证模式 180
10.2 建立和管理用户账号 181
10.2.1 Windows NT 认证模式登录账号的建立与删除 181
10.2.2 SQL Server 认证模式登录账号的建立与删除 181
10.2.3 通过企业管理器创建和删除登录账号 182
10.3 服务器角色与数据库角色 183
10.3.1 固定服务器角色 184
10.3.2 固定数据库角色 185
10.3.3 用户自定义数据库角色 194
习题 194

第 11 章 游标和其他概念 196

11.1 游标 196
11.2 游标定义 196
11.3 游标的使用及嵌套 198
11.3.1 打开并填充游标 198
11.3.2 读取游标 199
11.3.3 使用游标修改数据 200
11.3.4 关闭游标 200
11.3.5 释放游标 201
11.3.6 游标的嵌套 201
11.4 事务 202
11.4.1 事务的基本概念 202
11.4.2 显式事务 204
11.4.3 隐式事务 204
11.4.4 自动事务模式 204
11.4.5 分布式事务 205
11.5 锁定 205
11.5.1 锁定粒度 205
11.5.2 锁模式 205
11.5.3 显示锁定信息 206
11.5.4 死锁及其处理 207
11.6 监测 208
11.6.1 系统性能评估及监测 208
11.6.2 性能监测工具 209
习题 211

第 12 章 Visual Basic/SQL Server 开发与编程 …… 212

12.1 ODBC 概述 …… 212
12.1.1 ODBC 体系结构 …… 212
12.1.2 配置 ODBC 数据源 …… 213
12.2 OLE DB 和 ADO 概述 …… 216
12.2.1 OLE DB 体系结构 …… 217
12.2.2 ADO 对象模型 …… 218
12.2.3 使用 ADO 的简单流程 …… 219
12.3 使用 ADO 数据控件访问 SQL Server 数据库 …… 219
12.3.1 在 Visual Basic 工程中添加 ADO 数据控件 …… 220
12.3.2 使用 ADO 数据控件连接 SQL Server 数据库 …… 220
12.3.3 ADO 数据控件在程序中的操作方法 …… 224
12.3.4 数据绑定控件的使用 …… 226
12.4 使用 ADO 对象访问 SQL Server 数据库 …… 233
12.4.1 引用 ADO 对象库 …… 233
12.4.2 使用 Connection 对象 …… 234
12.4.3 使用 Recordset 对象处理数据 …… 236
12.4.4 使用 Command 对象 …… 244
习题 …… 249

第 13 章 学生信息管理系统 …… 250

13.1 需求分析 …… 250
13.2 系统设计 …… 250
13.2.1 系统功能分析 …… 250
13.2.2 系统功能模块设计 …… 250
13.3 数据库设计 …… 251
13.3.1 数据库需求分析 …… 251
13.3.2 数据库概念结构设计 …… 252
13.3.3 数据库逻辑结构设计 …… 253
13.3.4 数据结构的实现 …… 253
13.4 用户登录模块的创建 …… 254
13.5 学生信息管理系统主窗体的创建 …… 256
13.5.1 创建学生信息管理系统的主窗体 …… 256
13.5.2 主窗体内主要操作的程序实现 …… 257
13.5.3 创建公用模块 …… 260
13.6 子模块的编制 …… 262
13.6.1 添加学生基本信息 …… 262
13.6.2 修改删除学生基本信息 …… 265

13.6.3　查询学生基本信息 …… 267
13.7　系统编译与发行 …… 268
习题 …… 268

附录 …… 269

附录 A　数据库表结构 …… 269
附录 B　常用语句 …… 272
附录 C　常用函数 …… 273
附录 D　@@类函数 …… 278
附录 E　系统存储过程 …… 279
附录 F　扩展存储过程 …… 281

参考文献 …… 282

需要全本请访问 www.ertongbook.com

第 1 章

数据库基础概述

数据库技术是现代信息科学与技术的重要组成部分，是计算机数据处理与信息管理体制系统的核心。数据库技术研究解决计算机信息处理过程中大量数据有效地组织和存储的问题，在数据库系统中减少数据存储冗余，实现数据共享，保障数据安全以及高效地检索数据和处理数据。

随着计算机技术与网络通信技术的发展，数据库技术已成为信息社会中对大量数据进行组织与管理的重要技术手段及软件技术，是网络信息化管理系统的基础。本章主要介绍数据库系统的发展与组成，数据库的模型，数据库系统的网络结构，以及关系型数据库的分析与设计等内容，是学习和掌握现代数据库技术的基础。

1.1 数据库系统简介

随着计算机技术的发展，计算机的主要功能已从科学计算转变为事务处理。据统计，目前全世界 80%以上的计算机主要从事事务处理工作。在进行事务处理时，并不要求复杂的科学计算，主要是从大量有关数据中提取所需信息。因此，在进行事务处理时，必须在计算机系统中存入大量数据。为了有效地使用存放在计算机系统中的大量有关数据，必须采用一整套严密合理的存取数据、使用数据的方法。

数据管理是指对数据的组织、存储、维护和使用等。随着计算机技术的发展，数据管理的方法也在发展，大体上可将其分为三个阶段：人工管理阶段、文件管理阶段和数据库系统阶段。

人工管理阶段大致在 20 世纪 50 年代中期之前。那时，数据包含在程序中，用户必须考虑存储、使用数据的一切工作。因此，该阶段的数据管理是最低级的数据管理。

文件管理阶段大致从 20 世纪 50 年代后期开始，至 60 年代末。它是通过文件系统来管理和使用数据的。用户不必考虑数据在计算机系统中的实际存储方法（称为物理结构），只须考虑数据间的关系（称为逻辑结构）。文件系统中的文件属于个别程序所有。因此，文件管理阶段比人工管理阶段有了进步。但是，它仍然存在很严重的缺陷。例如，数据不能为有关用户共享，数据和程序间有着很大的依赖性等，从而造成数据的重复，冗余度既大，又极易造成数据的不一致性（修改数据时，相同重复数据未同时全部得到修改）。为了克服文件系统的缺点，产生了数据库系统。

数据库系统阶段始于 20 世纪 70 年代初，至今仍在发展。它的出现，标志着数据管理达到了一个全新的高级阶段。

1.2 数据库系统的组成

一个数据库系统（Database System）一般是由数据库（Database）、数据库管理系统（Database Management System，DBMS）以及数据库用户组成。广义地说，数据库系统是由计算机硬件、操作系统、数据库管理系统以及在它支持下建立起来的数据库、数据库应用程序、用户和维护人员组成的一个整体。

1.2.1 数据库

1．数据库的基本概念

数据库是存储在计算机系统内的有结构的数据的集合。通俗地讲，这些数据是被数据库管理系统按一定的组织形式存放在各个数据库文件中的。也就是说，数据库是由很多数据库文件以及若干辅助操作文件组成的。存放在数据库中的数据可以被所有合法用户使用。

2．数据库的基本模型

基本的数据库模型有三种：网状数据库模型、层次数据库模型和关系型数据库模型。网状数据库模型是将每个记录当成一个节点，节点和节点之间可以建立关联，形成一个网状结构。层次型数据库模型是采用树状结构，依据数据的不同类型，将数据分门别类，存储在不同的层次之下。关系型数据库模型是以二维矩阵来存储数据的，行和列形成一个关联的数据表。

现在经常使用的数据库系统产品几乎都是关系型的，包括微软的 SQL Server、IBM 的 ORACLE、SYBASE 等。另外，还有一些小型数据库管理系统如 Access、FoxPro 和 PowerBuild 等。

1.2.2 数据库管理系统

数据库管理系统是指在操作系统的支持下帮助用户建立、使用和管理数据库的软件系统。通常包含数据描述语言、数据操作语言以及管理和控制程序三个组成部分。

1．数据描述语言（Data Description Language）

用来描述数据库的结构，供用户建立数据库。

2．数据操作语言（Data Manipulation LangUage，DML）

供用户对数据库进行数据的查询（数据的检索和统计等）和处理（数据的增加、删除和修改等）等操作。

3．管理和控制程序

包括安全、通信控制和工作日志等。

1.2.3 数据库系统的用户

1．系统程序员

系统程序员负责整个数据库系统的设计工作，依据用户的需求安装数据库管理系统，建立维护数据库管理系统及相关软件的工具，设计合适的数据库及表文件，并对整个数据库的存取权限作出规划。

2．数据库管理员

数据库管理员（Database Administrator, DBA）是支持数据库系统的专业技术人员。数据库管理员的任务主要是决定数据库的内容，对数据库中的数据进行修改、维护，对数据库的运行状况进行监督，并且管理账号，备份和还原数据，以及提高数据库的运行效率。

3．应用程序员

应用程序员负责编写访问数据库的面向终端用户的应用程序，使用户可以很友好地使用数据库。可以使用 Visual Basic、Delphi、PHP、ASP 和 JSP 等来开发数据库应用程序。

4．操作员

操作员（普通用户）只需操作应用程序软件来访问数据库，利用数据库系统完成日常的工作，不关心数据库的具体格式及其维护和管理等问题。

1.2.4　数据库系统的网络结构

1．Main Frame 大型数据库

大型数据库是由一台性能很强的计算机（称为主机或者数据库服务器）负责处理庞大的数据，用户通过终端机与大型主机相连，以存取数据。

2．本地小型数据库

在用户较少、数据量不大的情况下，可使用本地小型数据库。小型数据库一般是由个人建立的个人数据库。常用的 DBMS 有 Access 和 FoxPro 等。

3．分布式数据库

分布式数据库是为了解决大型数据库反应缓慢的问题而提出的，它是由多台数据库服务器组成。数据可来自不同的服务器。

4．客户机/服务器数据库

在客户机/服务器数据库的网络结构中，数据库的处理可分为两个系统，即客户机（Client）和数据库服务器（Database Server），前者运行数据库应用程序，后者运行全部或者部分数据库管理系统。在客户机上的数据应用程序将该请求通过网络发送给服务器，数据库服务器进行搜索，并将用户查询所需的数据返回到客户机。

1.3　关系型数据库

1.3.1　关系型数据库定义

关系型数据库概念是由 E．F．Codd 博士提出的。1976 年 6 月他发表了《关于大型共享数据库数据的关系模型》的论文，在论文中他阐述了关系数据库模型及其原理，并把它用于数据库系统中。

数据库系统的发展经历了三个阶段：网状数据库、层次数据库和关系型数据库。而关系型数据库之所以能被广泛的应用,是因为它将每个具有相同属性的数据独立地存储在一个表中。它解决了层次型数据库的横向关联不足的缺点，也避免了网状数据库关联过于复杂的问题。

关系型数据库是指一些相关的表和其他数据库对象的集合。在关系型数据库中，信息存

放在二维表格结构的表中，一个关系型数据库包含多个数据表，每一个表包含行（记录）和列（字段）。一般来说，关系型数据库都有多个表。关系型数据库所包含的表之间是有关联的，关联性由主键、外键所体现的参照关系实现。关系型数据库不仅包含表，还包含其他数据库对象，例如关系图、视图、存储过程和索引等。

1.3.2 关系型数据库与表

关系型数据库是由多个表和其他数据库对象组成的。表是一种最基本的数据库对象，类似于电子表格，是由行和列组成的，除第一行（表头）以外，表中的每一行通常称为一条记录，表中的每一列称为一个字段，表头的各列给出了各个字段的名称。例如，图 1-1 中的学生信息表中收集了一些同学的个人资料,这些资料就可以用关系型数据库中的一个表来存储。其中“ student_Name ”(姓名)、“ address ”(家庭住址)、“ tele_Number ”(电话号码）等为表的字段，如果要查找“ 张小强 ”的家庭住址，则可由“ 张小强 ”所在的行与字段“ address ”所在的列关联相交处得到。

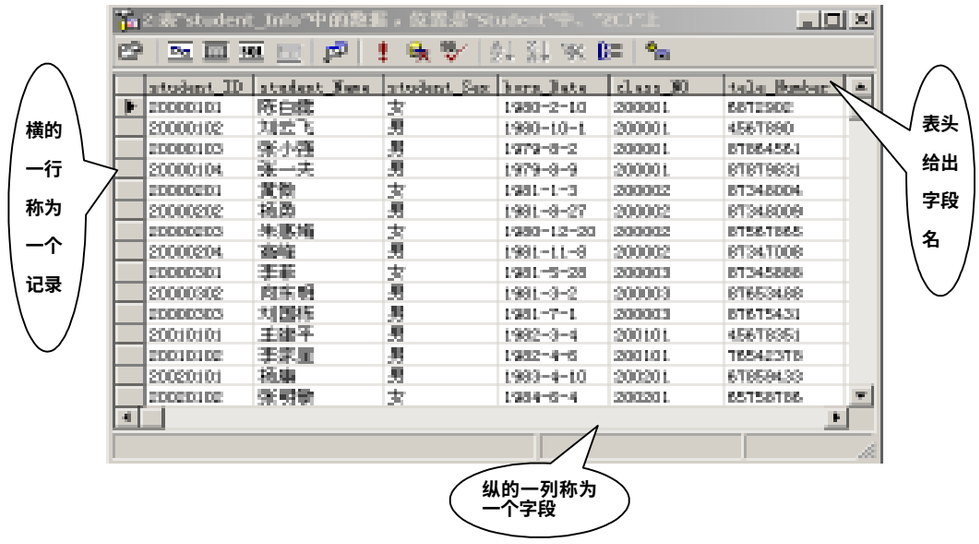


图 1-1 学生信息表

在关系型数据库中，如果有多个表存在，则表与表之间也会因为字段的关系而产生关联。

1.3.3 主键与外键

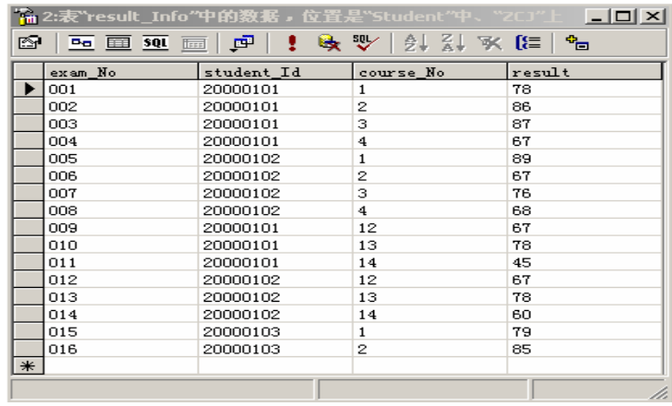
关系型数据库中的一个表是由行和列组成的，要求表中的每行记录都必须是惟一的，而不允许出现完全相同的记录。在设计表时，可以通过定义主键（PRIMARY KEY）来保证记录（实体）的惟一性。

一个表的主键由一个或多个字段组成，其值具有惟一性，而且不允许取空值（NULL），主键的作用是惟一地标识表中的每一条记录。例如，在一个“ 学生表 ”中可用“ 学号 ”字段作为主键，但不能使用“ 姓名 ”字段作为主键，因为同名同姓的现象还是屡见不鲜的。为了惟一地标识实体的每一个实例，每个数据库表都应当有一个主键，而且只能有一个主键。有

时表中可能没有一个字段具有惟一性，没有任何字段可以作为表的主键。在这种情况下，可以考虑使用两个或两个以上字段的组合作为主键。

一个关系型数据库可能包含多个表，可以通过外键（FOREIGN KEY）使这些表之间关联起来。如果在表 A 中有一个字段对应于表 B 中的主键，则该字段称为表 A 的外键。虽然该字段出现在表 A 中，但由它所标识的主体的详细信息却存储在表 B 中，对于表 A 来说这些信息就是存储在表的外部，故称之为外键。

如图 1-2 所示的学生考试“成绩表”中有两个外键，一个是学号“student_Id”，其详细信息存储在“学生表”中；另一个是课程编号“course_No”，其详细信息存储在“课程表”中。“成绩表”和“学生表”各有一个“学号”字段，该字段在“成绩表”中是外键，在“学生表”中则是主键，但这两个字段的数据类型以及字段宽度必须完全一样，字段的名称可以相同，也可以不相同。



exam_No	student_Id	course_No	result
001	20000101	1	78
002	20000101	2	86
003	20000101	3	87
004	20000101	4	67
005	20000102	1	89
006	20000102	2	67
007	20000102	3	76
008	20000102	4	68
009	20000101	12	67
010	20000101	13	78
011	20000101	14	45
012	20000102	12	67
013	20000102	13	78
014	20000102	14	60
015	20000103	1	79
016	20000103	2	85

图 1-2　主键与外键的关系

1.3.4　字段约束

设计表时，可对表中的一个字段或多个字段的组合设置约束条件，让 SQL Server 检查该字段的输入值是否符合这个约束条件。约束分为表级约束和字段级约束两种。表级约束是一个表中几个字段的约束，字段级约束则是对表中一个字段的约束。下面介绍几种常见的约束形式。

1．PRIMARY KEY

PRIMARY KEY 用来保证表中每条记录的惟一性。设计一个表时，可用一个字段或多个字段（最多 16 个字段）的组合作为这个表的主键。用单个字段作为主键时，使用字段级约束；用字段组合作为主键时，则使用表级约束。

每个表只能有一个主键。如果不在主键字段中输入数据，或输入的数据在前面已经输入过，则这条记录将被拒绝。

2．FOREIGN KEY

FOREIGN KEY 字段与其他表中的主键字段或具有惟一性的字段相对应，其值必须在所引用的表中存在，而且所引用的表必须存放在同一关系型数据库中。如果在外键字段中输入

一个非 NULL 值，但该值在所引用的表中并不存在，则这条记录也会被拒绝，因为这样将破坏两表之间的关联性。外键字段本身的值不要求是惟一的。

3．NULL 与 NOT NULL

若在一个字段中允许不输入数据，则可以将该字段定义为 NULL，如果在一个字段中必须输入数据，则应当将该字段定义为 NOT NULL。一个字段中出现 NULL 值意味着用户还没有为该字段输入值，NULL 值既不等价于数值型数据中的 0，也不等价于字符型数据中的空字符串。

4．UNIQUE

如果一个字段值不允许重复，则应当对该字段添加 UNIQUE 约束。与主键不同的是，在 UNIQUE 字段中允许出现 NULL 值，但为保持惟一性，最多只能出现一次 NULL 值。

5．CHECK

CHECK 约束用于检查一个字段或整个表的输入值是否满足指定的检查条件。在表中插入或修改记录时，如果不符合这个检查条件，则这条记录将被拒绝。

6．DEFAULT

DEFAULT 约束用于指定一个字段的默认值，当尚未在该字段中输入数据时，该字段中将自动填入这个默认值。若对一个字段添加了 NOT NULL 约束，但又没有设置 DEFAULT 约束，就必须在该字段中输入一个非 NULL 值，否则将会出现错误。

1.3.5 数据完整性

数据完整性用于保证关系型数据库中数据的正确性和可靠性。如果有一个学生以学号 20000101 登记注册，则关系型数据库中就不能再有其他学生使用相同的学号。如果学生成绩是用百分制来表示的，则关系型数据库中就不能接受小于 0 或超过 100 的值。如果“学生表”中有一个“专业”字段，则该字段的值就只能允许学校已经开设的专业名称或编号。

规划关系型数据库表时有两个重要步骤：其一是如何确定一个字段的有效值，其二是决定如何强制实施字段的数据完整性。数据完整性分为以下 4 种类型。

1．实体完整性

实体完整性（Entity Integrity）用于保证关系型数据库表中的每一条记录都是惟一的，建立主键的目的就是为了实施实体完整性。一个表中的主键不能取空值，也不能取重复的值。例如，选择“学号”字段作为“学生表”中的主键时，每一条记录中的“学号”字段值就应输入一个非空值，而且必须是各不相同的。

2．域完整性

域完整性（Domain Integrity）用于保证给定字段中数据的有效性，即保证数据的取值在有效的范围内。例如，限制“成绩”字段的值是在 0 到 100 之间，如果在该字段中输入小于 0 或大于 100 的值，就破坏了该字段的域完整性。又如，在“成绩表”中“学号”字段是一个外键，该字段的值只能是“学生表”已经存在的学号，如果在该字段中输入“学生表”所没有的学号，也将破坏该字段的域完整性。

3．参照完整性

参照完整性（Referential Integrity）是用于确保相关联的表间的数据保持一致。当添加、删除或修改关系型数据库表中的记录时，可以借助于参照完整性来保证相关联的表之间的数

需要全本请访问 www.ertongbook.com