高等学校计算机科学与技术教材

数据库应用系统开发实例

主　编　曹红根　丁　勇
副主编　姜　枫　袁宝华　高广银　钱　鸣
主　审　朱辉生

清 华 大 学 出 版 社
北京交通大学出版社

·北京·

需要全本请访问 www.ertongbook.com

内 容 简 介

本书精选了5种当前应用较广泛的开发工具，以学生综合信息管理系统为背景，详细讲述了使用SQL Server 2000开发该系统的全过程。本书共分为9章。第1章到第3章介绍了使用SQL Server进行数据库系统开发的一些基础性知识。第4章针对学生综合信息管理系统，介绍了数据库设计的全过程。第5章到第9章分别介绍了使用各种开发工具开发学生综合信息管理系统的全过程。

本书不仅对高校计算机等相关专业的学生进行毕业设计、课程设计具有较高的参考价值，还适合广大使用SQL Server进行数据库系统开发的软件开发人员参考。

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　13501256678　13801310933

图书在版编目（CIP）数据

数据库应用系统开发实例/曹红根，丁勇主编. —北京：清华大学出版社；北京交通大学出版社，2008.9
（高等学校计算机科学与技术教材）
ISBN 978-7-81123-359-9

Ⅰ.数… Ⅱ.①曹… ②丁… Ⅲ.数据库系统-系统开发-高等学校-教材 Ⅳ.TP311.13

中国版本图书馆CIP数据核字（2008）第114162号

责任编辑：杨正泽
出版发行：清华大学出版社　邮编：100084　电话：010-62776969　http://www.tup.com.cn
　　北京交通大学出版社　邮编：100044　电话：010-51686414　http://press.bjtu.edu.cn
印 刷 者：北京东光印刷厂
经　　销：全国新华书店
开　　本：185×230　印张：28.25　字数：630千字
版　　次：2008年9月第1版　2008年9月第1次印刷
书　　号：ISBN 978-7-81123-359-9/TP·426
印　　数：1～4 000册　定价：42.00元

本书如有质量问题，请向北京交通大学出版社质监组反映。对您的意见和批评，我们表示欢迎和感谢。
投诉电话：010-51686043，51686008；传真：010-62225406；E-mail：press@bjtu.edu.cn。

前　言

数据库技术是高校计算机科学与技术、信息管理与信息系统等相关专业重要的专业方向之一，学生课程设计、毕业设计及就业方向大都与数据库技术相关，因此加强学生的数据库应用能力培养尤为重要。*SQL Server 2000* 作为 *Microsoft* 公司在数据管理领域精心打造的产品，具有高度的可伸缩性和可靠性，提供了对 *Web* 的完全支持，并拥有强大的集成和可扩展的分析功能，能够帮助用户进行快速开发、调试和数据转换。因此，基于 *SQL Server 2000* 的数据库应用程序开发必将成为今后的一个主要方向。

本书从实战着手，使用了当今最流行的几种开发工具，包括 *Visual Basic*、*Visual C++*、*PowerBuilder*、*ASP*、*J2EE* 等，以数据库应用开发中极具代表性的案例——学生综合信息管理系统为背景，详细讲述了使用 *SQL Server 2000* 开发信息管理系统的全过程，包括案例的需求分析、系统建模、系统配置、数据库分析、界面设计和代码分析与实现等，以使读者能够透彻地掌握信息管理系统的开发方法和步骤，开发出具有实用价值的信息管理系统。

本书共分 *9* 章。第 *1* 章到第 *3* 章介绍了使用 *SQL Server* 进行数据库系统开发的一些基础性知识，包括数据库基础、*SQL Server* 关系数据库系统、*Transact－SQL* 语言基础、*SQL Server* 数据库对象管理、数据库设计等方面的详细介绍。第 *4* 章针对学生综合信息管理系统，介绍了数据库设计的全过程。第 *5* 章到第 *9* 章是本书的精华部分，分别介绍了使用 *Visual Basic*、*Visual C++*、*PowerBuilder*、*ASP*、*J2EE* 等开发工具开发学生综合信息管理系统的全过程。

考虑到部分读者对各种编程语言以 *SQL Server* 为后台数据库的信息管理系统的开发不是很了解，所以本书在详细介绍具体的系统开发之前，都有侧重点地讲解了所用语言的数据库编程技术，以及与 *SQL Server* 相关的数据库技术，这些技术都具有相当的实用性和代表性，是信息管理系统必须掌握的技术。

本书不仅对高校计算机等相关专业的学生进行毕业设计、课程设计具有很高的参考价值，还适合广大使用 *SQL Server* 进行数据库系统开发的软件开发人员参考。

本书由朱辉生主审，第 *5* 章由曹红根编写，第 *1*、*7* 章由丁勇编写，第 *2*、*3*、*4* 章由高广银编写，第 *6* 章由袁宝华编写，第 *8* 章由钱鸣编写，第 *9* 章由姜枫编写。

由于时间仓促及作者水平有限，书中遗漏及不妥之处在所难免，恳请广大读者批评指正。

编　者

2008 年 8 月

目　　录

第 1 章　数据库基本原理 …… 1
1.1　数据库系统概述 …… 1
1.1.1　数据库的地位 …… 1
1.1.2　四个基本概念 …… 1
1.1.3　数据库系统结构 …… 2
1.2　关系模型 …… 4
1.2.1　关系数据结构 …… 4
1.2.2　关系操作 …… 4
1.2.3　关系完整性约束 …… 5
1.3　SQL 语言概述 …… 6
1.3.1　SQL 概述 …… 6
1.3.2　SQL 语言的组成 …… 6
1.3.3　SQL 语句的结构 …… 7
1.3.4　常用的 SQL 语句 …… 7
第 2 章　数据库系统设计 …… 13
2.1　数据库设计概述 …… 13
2.2　需求分析 …… 13
2.2.1　需求分析的任务 …… 13
2.2.2　需求分析的方法 …… 14
2.2.3　数据字典 …… 15
2.3　概念结构设计 …… 17
2.3.1　概念结构 …… 17
2.3.2　概念设计的方法与步骤 …… 17
2.3.3　数据抽象与局部视图设计 …… 20
2.3.4　视图的集成 …… 22
2.4　逻辑结构设计 …… 26
2.4.1　E－R 图向关系模型的转换 …… 27
2.4.2　数据模型的优化 …… 28

第 3 章　SQL Server 2000 数据库系统 …… 30
3.1　SQL Server 2000 简介 …… 30
3.2　SQL Server 企业管理器 …… 31
3.2.1　企业管理器的环境 …… 31
3.2.2　启动与关闭服务器 …… 32
3.2.3　企业管理器的常见用途 …… 34
3.3　SQL Server 查询分析器 …… 34
3.3.1　SQL Server 查询分析器简介 …… 34
3.3.2　配置 SQL Server 查询分析器 …… 35
3.3.3　使用 SQL Server 查询分析器 …… 36
3.4　SQL Server 2000 的数据库创建 …… 39
3.4.1　文件与文件组 …… 39
3.4.2　用企业管理器创建数据库 …… 40
3.4.3　创建数据库表 …… 43
3.4.4　创建和使用约束 …… 45
第 4 章　学生综合信息管理系统的研究 …… 48
4.1　概述 …… 48
4.1.1　设计原则 …… 48
4.1.2　需求分析 …… 49
4.2　数据库的设计 …… 55
4.2.1　概念结构设计 …… 55
4.2.2　逻辑结构设计 …… 58
4.3　系统的功能结构 …… 60
4.3.1　管理员子系统 …… 61
4.3.2　教师子系统 …… 61
4.3.3　学生子系统 …… 62
第 5 章　基于 Visual Basic 的系统实现 …… 63
5.1　常用数据库控件介绍 …… 63
5.1.1　DataGrid 控件 …… 63
5.1.2　MSFlexGrid 控件 …… 68
5.2　使用 ADO 访问数据库 …… 70
5.2.1　ADO 概述 …… 70
5.2.2　ADO 对象模型简介 …… 71
5.2.3　ADO 的编程模型 …… 71
5.2.4　在 Visual Basic 中使用 ADO 对象 …… 72

需要全本请访问 www.ertongbook.com

5.3 创建学生综合信息管理系统工程 …… 79
5.4 创建公用模块 …… 82
5.4.1 公用模块的功能 …… 82
5.4.2 公用模块的设计 …… 82
5.5 创建启动窗体模块 …… 84
5.5.1 窗体界面设计 …… 84
5.5.2 启动窗体的设置 …… 84
5.5.3 编写窗体代码 …… 85
5.6 创建登录模块 …… 86
5.7 创建主窗体模块 …… 89
5.7.1 主窗体菜单设计 …… 89
5.7.2 主窗体工具栏设计 …… 90
5.7.3 主窗体界面设计 …… 92
5.7.4 主窗体状态栏设计 …… 93
5.7.5 编写窗体代码 …… 94
5.8 实现数据报表功能 …… 99
5.8.1 数据环境设计器 …… 99
5.8.2 建立报表 …… 102
5.9 创建学生信息查询模块 …… 105
5.9.1 学生基本信息查询窗体设计 …… 106
5.9.2 学生成绩查询窗体设计 …… 107
5.10 创建教师用户模块 …… 109
5.10.1 教师信息查询窗体设计 …… 109
5.10.2 教师授课课程成绩维护窗体设计 …… 111
5.11 创建系统用户模块 …… 121
5.11.1 学生基本信息维护窗体设计 …… 121
5.11.2 学生选课信息维护窗体设计 …… 131
5.11.3 教师基本信息维护窗体设计 …… 133
5.11.4 教师授课信息维护窗体设计 …… 135
5.12 创建其他模块 …… 139
5.12.1 用户密码修改窗体设计 …… 139
5.12.2 关于窗体设计 …… 141
5.12.3 实现系统退出功能 …… 141
5.12.4 实现系统重新登录功能 …… 142
5.13 系统的改进与扩展 …… 143

第 6 章　基于 Visual C++ 的系统实现 …… 144
6.1　用 Visual C++开发数据库程序概述 …… 144
6.1.1　ODBC 技术 …… 144
6.1.2　DAO 技术 …… 144
6.1.3　ADO 技术 …… 144
6.2　常用数据库控件介绍 …… 146
6.2.1　ADO Data 控件 …… 147
6.2.2　Data Grid 控件 …… 153
6.2.3　DataCombo 控件 …… 155
6.3　创建项目框架 …… 158
6.3.1　新建工程 …… 159
6.3.2　设计 ADO 封装类 …… 160
6.3.3　添加 ADO 控件 …… 164
6.3.4　为表创建类 …… 164
6.4　创建登录窗体模块 …… 169
6.5　创建主窗体模块 …… 174
6.5.1　主窗体菜单设计 …… 174
6.5.2　主窗体界面设计 …… 177
6.5.3　主窗体状态栏设计 …… 182
6.6　创建学生用户模块 …… 183
6.6.1　学生信息查询窗体设计 …… 184
6.6.2　学生成绩查询窗体设计 …… 186
6.7　创建教师用户模块 …… 188
6.7.1　教师信息查询窗体设计 …… 188
6.7.2　教师授课课程成绩维护窗体设计 …… 191
6.8　创建系统用户模块 …… 197
6.8.1　学生基本信息维护窗体设计 …… 197
6.8.2　学生选课信息维护窗体设计 …… 204
6.8.3　教师基本信息维护窗体设计 …… 209
6.8.4　教师授课信息维护窗体设计 …… 213
6.9　创建公共模块 …… 216
第 7 章　基于 PowerBuilder 的系统实现 …… 219
7.1　常用数据库控件介绍 …… 219
7.1.1　数据窗体对象 …… 219
7.1.2　数据窗体控件 …… 231

7.2 数据库连接方法 …… 234
7.3 系统实现 …… 239
7.3.1 登录界面和主界面设计 …… 239
7.3.2 学生信息管理界面设计 …… 246
7.3.3 教师信息管理界面设计 …… 255
7.3.4 系部和班级管理界面设计 …… 262
7.4 开发技巧与难点设计 …… 265
7.5 系统发布 …… 277
第 8 章 基于 ASP 的系统实现 …… 279
8.1 搭建 ASP 开发环境 …… 279
8.1.1 安装 IIS …… 279
8.1.2 启动 IIS 并配置网站 …… 281
8.1.3 使用 Dreamweaver 创建网站 …… 284
8.2 ASP 网页的基本结构 …… 288
8.3 ASP 内置对象 …… 290
8.4 编写公共模块文件 …… 293
8.4.1 数据库连接文件 …… 293
8.4.2 样式表文件 …… 296
8.4.3 页面尾文件 …… 297
8.4.4 框架页面和页面头文件设计 …… 298
8.4.5 公共函数的编写（function.asp） …… 300
8.4.6 用户登录页面 …… 302
8.4.7 用户注销页面 …… 305
8.4.8 用户登录判定 …… 306
8.5 系统管理员子模块的实现 …… 306
8.5.1 学生/教师基本信息维护页面 …… 306
8.5.2 教师授课信息维护页面 …… 319
8.5.3 学生选课信息维护页面 …… 328
8.5.4 修改密码页面 …… 332
8.6 教师子模块的实现 …… 333
8.6.1 信息浏览页面 …… 333
8.6.2 成绩录入页面 …… 337
8.6.3 成绩修改页面 …… 344
8.7 学生子模块的实现 …… 345
8.7.1 基本信息浏览页面 …… 346

8.7.2 选课成绩浏览页面 …… 348
第9章 基于J2EE的系统实现 …… 352
9.1 环境配置 …… 352
9.1.1 JDK的安装与配置 …… 352
9.1.2 Tomcat的安装与配置 …… 353
9.1.3 Eclipse与MyEclipse的安装与配置 …… 354
9.1.4 使用MyEclipse进行简单Web项目开发 …… 355
9.2 数据库连接方法 …… 358
9.2.1 JDBC简介 …… 358
9.2.2 使用JDBC连接SQL Server 2000 …… 359
9.3 Struts基础知识 …… 362
9.3.1 Struts简介 …… 362
9.3.2 使用MyEclipse开发Struts简单应用 …… 364
9.4 Hibernate基础知识 …… 373
9.4.1 Hibernate简介 …… 373
9.4.2 使用MyEclipse开发Hibernate简单应用 …… 373
9.5 系统实现 …… 381
9.5.1 将数据库中各表利用Hibernate做ORM映射 …… 381
9.5.2 登录模块和主界面模块设计 …… 381
9.5.3 学生子系统模块设计 …… 388
9.5.4 教师子系统模块设计 …… 396
9.5.5 管理员子系统模块设计 …… 402
参考文献 …… 439

第 1 章

数据库基本原理

1.1 数据库系统概述

1.1.1 数据库的地位

数据库技术产生于 20 世纪 60 年代末，是数据管理的最新技术，是计算机科学的重要分支。数据库技术的出现极大地促进了计算机应用向各行各业的渗透。数据库的建设规模、数据库信息量的大小和使用频度已成为衡量一个国家信息化程度的重要标志。

1.1.2 四个基本概念

1. 数据（Data）

数据是数据库中存储的基本对象。可以对数据作如下定义：描述事物的符号记录。其种类可以是文字、图形、图像、声音、语言等。其特点是数据与其语义不可分。例如：学生档案中的学生记录用数据（张三，男，1980，江苏，计算机科学系）来描述。其语义为：学生姓名、性别、出生年月、籍贯、所在系别，解释成张三是个男生，1980 年出生，江苏人，在计算机科学系学习。

2. 数据库（Database，简称 DB）

数据库是指长期储存在计算机内、有组织的、可共享的大量数据的集合。数据库的特征：①数据按一定的数据模型组织、描述和储存；②可为各种用户共享；③冗余度较小；④数据独立性较高，易扩展。

3. 数据库管理系统（DBMS）

数据库管理系统是位于用户与操作系统之间的一层数据管理软件，是数据库系统的一个重要组成部分，用来科学地组织和存储数据、高效地获取和维护数据。其主要功能有如下几种。

① 数据定义功能，提供数据定义语言（DDL），用来定义数据库中的数据对象。

② 数据操纵功能，提供数据操纵语言（DML），用来操纵数据，实现对数据库的基本操作，包括查询、插入、删除和修改等。

③ 数据库的运行管理，保证数据的安全性、完整性，并发控制和故障恢复。

④ 数据库的建立和维护功能，是一组实用程序完成数据库数据的批量装载、数据库转储、介质故障恢复、数据库的重组织和性能监视等。

4. 数据库系统（DBS）

数据库系统是指在计算机系统中引入数据库后的系统构成。在不引起混淆的情况下常常把数据库系统简称为数据库。由数据库、数据库管理系统（及其开发工具）、应用系统、数据库管理员（和用户）构成。

1.1.3 数据库系统结构

从数据库管理系统角度看，数据库系统通常采用三级模式结构，这是数据库管理系统内部的模式结构。从数据库最终用户角度看，数据库系统的结构分为集中式结构、分布式结构、客户/服务器结构等。

1. 数据库系统模式的概念

模式（Schema）是数据库逻辑结构和特征的描述，它反映的是数据的结构。模式是相对稳定的，模式的一个实例（Instance）是模式的一个具体值，它反映数据库某一时刻的状态，同一个模式可以有很多实例，实例随数据库中的数据的更新而变动。

2. 数据库系统的三级模式结构

数据库系统的三级模式结构是指数据库系统是由外模式、模式和内模式三级构成，如图 1－1 所示。

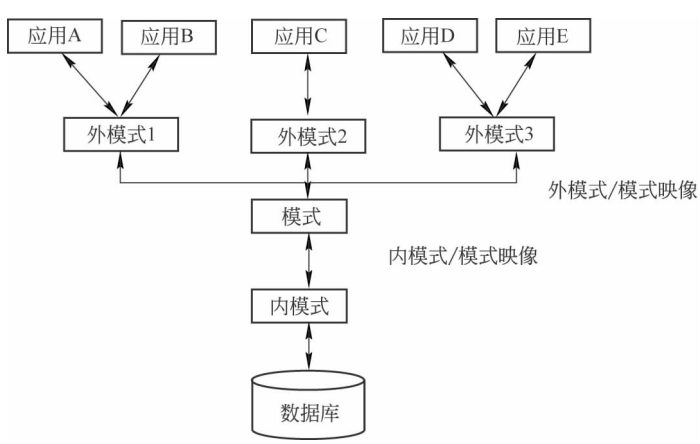


图 1－1 三级模式结构

需要全本请访问 www.ertongbook.com

（1）**模式**（Schema）

模式也称逻辑模式，是数据库中全体数据的逻辑结构和特征的描述，是所有用户的公共数据视图，一个数据库只有一个模式。模式是数据库系统模式结构的中间层，既不涉及数据的物理存储细节和硬件环境，也与具体的应用程序及高级程序设计语言无关。

（2）**外模式**（External Schema）

外模式也称子模式或用户模式，是数据库用户能够看见和使用的局部数据的逻辑结构和特征的描述，是数据库用户的数据视图，是与某一应用有关的数据的逻辑表示。

外模式通常是模式的子集，一个数据库可以有多个外模式，反映了不同的用户的应用需求、看待数据的方式、对数据保密的要求。对模式中同一数据，在外模式中的结构、类型、长度、保密级别等都可以不同。同一外模式也可以为某一用户的多个应用系统所使用，但一个应用程序只能使用一个外模式。外模式是保证数据库安全性的一个有力措施，每个用户只能看见和访问所对应的外模式中的数据。

（3）**内模式**（Internal Schema）

内模式也称存储模式，是数据物理结构和存储方式的描述，是数据在数据库内部的表示方式。例如记录的存储方式是顺序存储，还是堆存储，是 B＋树索引，还是 hash 索引等。一个数据库只有一个内模式。

3. 数据库的二级映像功能与数据独立性

三级模式是对数据的 3 个抽象级别，二级映像在 DBMS 内部实现这三个抽象层次的联系和转换，数据库系统的三级模式结构包括以下两种。

（1）外模式/模式映像

外模式/模式映像定义了外模式与模式之间的对应关系。每一个外模式都对应一个外模式/模式映像。映像定义通常包含在各自外模式的描述中。

外模式/模式映像保证了数据的逻辑独立性，当模式改变时，数据库管理员修改有关的外模式/模式映像，使外模式保持不变，应用程序是依据数据的外模式编写的，从而应用程序不必修改，保证了数据与程序的逻辑独立性。

（2）模式/内模式映像

模式/内模式映像定义了数据全局逻辑结构与存储结构之间的对应关系。例如，说明逻辑记录和字段在内部是如何表示的，数据库中模式/内模式映像是唯一的，该映像定义通常包含在模式描述中。

模式/内模式映像保证了数据的物理独立性，当数据库的存储结构改变了，例如选用了另一种存储结构，数据库管理员修改模式/内模式映像，使模式保持不变，应用程序不受影响。保证了数据与程序的物理独立性。

1.2 关系模型

关系数据库系统是支持关系模型的数据库系统。关系模型的组成包括关系数据结构、关系操作、关系完整性约束。

1.2.1 关系数据结构

关系模型的数据结构非常单一，现实世界的实体及实体间的各种联系均可以用关系来表示。从用户角度，关系模型中数据的逻辑结构是一张二维表，它由行和列组成。以下从集合论角度给出关系数据结构的形式化定义。

- 关系（Relation）：一个关系对应一张二维表。
- 元组（Tuple）：表中的一行即为一个元组。
- 属性（Attribute）：表中的一列即为一个属性，给每一个属性起一个名称（属性名）。
- 主码（Key）：表中的某个属性组，它可以唯一确定一个元组。
- 域（Domain）：属性的取值范围。
- 分量：元组中的一个属性值。
- 关系模式：对关系的描述，一般表示为：关系名（属性1，属性2，…，属性n）。

它可以形式化地表示为R（U，D，DOM，F），其中R为关系名，U为组成该关系的属性名集合，D为属性组U中属性所来自的域，DOM为属性向域的映像集合，F为属性间数据的依赖关系集合。

1.2.2 关系操作

关系模型给出了关系操作能力的说明，但不对RDBMS语言给出具体的语法要求，也就是说不同的RDBMS可以定义和开发不同的语言来实现这些操作。

1. 基本的关系操作

关系模型中基本的关系操作包括查询（Query）操作和插入（Insert）、删除（Delete）、修改（Update）操作两大部分。关系的查询表达能力很强，是关系操作中最主要的部分。查询操作又可以分为选择（Select）、投影（Project）、连接（Join）、除（Divide）、并（Union）、差（Except）、交（Intersection）等。

2. 关系数据语言的种类

分为关系代数和关系演算。关系代数是用对关系的运算来表达查询要求。关系演算是用

谓词来表达查询要求，关系演算又可按谓词变元的基本对象是元组变量还是域变量分为元组关系演算和域关系演算。关系代数、元组关系演算和域关系演算均是抽象的查询语言。

SQL 是具有关系代数和关系演算双重特点的语言，集查询、DDL、DML 和 DCL 于一体。

1.2.3　关系完整性约束

关系模型的完整性规则是对关系的某种约束条件，关系模型中有三类完整性约束：实体完整性、参照完整性、用户定义的完整性。

1. 实体完整性

实体完整性规则（Entity Integrity）：若属性 A 是基本关系 R 的主属性，则属性 A 不能取空值。

① 实体完整性规则是针对基本关系而言的。一个基本表通常对应现实世界的一个实体集。

② 现实世界中的实体是可区分的，即它们具有某种唯一性标识。

③ 关系模型中以主码作为唯一性标识。

④ 主码中的属性即主属性不能取空值。空值就是“不知道”或“无意义”的值。主属性取空值，说明存在某个不可标识的实体，即存在不可区分的实体。

2. 参照完整性

在关系模型中实体及实体间的联系都是用关系来描述的，因此可能存在着关系与关系间的引用。设 F 是基本关系 R 的一个或一组属性，但不是关系 R 的码。如果 F 与基本关系 S 的主码 K_s 相对应，则称 F 是基本关系 R 的外码，基本关系 R 称为参照关系（Referencing Relation），基本关系 S 称为被参照关系（Referenced Relation）或目标关系（Target Relation）。

① 关系 R 和 S 不一定是不同的关系。

② 目标关系 S 的主码 K_s 和参照关系的外码 F 必须定义在同一个（或一组）域上。

③ 外码并不一定要与相应的主码同名。

④ 当外码与相应的主码属于不同关系时，往往取相同的名字，以便于识别。

例 1-1　学生实体、专业实体及专业与学生间的一对多联系。

专业（<u>专业号</u>，专业名）

学生（学号，姓名，性别，年龄，<u>专业号</u>）

这两个关系之间存在着属性的引用，即学生关系引用了专业关系的主码“专业号”。

例 1-2　学生、课程、学生与课程之间的多对多联系。

学生（<u>学号</u>，姓名，性别，年龄，专业号）

课程（<u>课程号</u>，课程名，学分）

选修（学号，课程号，成绩）

选课关系引用了学生关系的主码“学号”，同时引用了课程关系的主码“课程号”。不仅两个或两个以上的关系间可以存在引用关系，同一关系内部属性间也可能存在引用关系。

参照完整性规则：若属性（或属性组）F 是基本关系 R 的外码，它与基本关系 S 的主码 K_s 相对应（基本关系 R 和 S 不一定是不同的关系），则对于 R 中每个元组在 F 上的值必须为：

① 或者取空值（F 的每个属性值均为空值）；

② 或者等于 S 中某个元组的主码值。

例 1-3　学生关系中每个元组的“专业号”属性只取下面两类值：

① 空值，表示尚未给该学生分配专业；

② 非空值，这时该值必须是专业关系中某个元组的“专业号”值，表示该学生不可能分配到一个不存在的专业中。

3. 用户定义的完整性

用户定义的完整性是针对某一具体关系数据库的约束条件，它反映某一具体应用所涉及的数据必须满足的语义要求。如某个属性必须取唯一值、某个非主属性不能取空值、某个非主属性的取值有一定的范围等。

例 1-4　课程（课程号，课程名，学分），“课程号”属性必须取唯一值，非主属性“课程名”不能取空值，非主属性“学分”只能取值 {1，2，3，4}。

1.3　SQL 语言概述

SQL 语言是一个通用的、功能极强的关系数据库语言。

1.3.1　SQL 概述

SQL 语言是 1974 年由 Boyce 和 Chamberlin 提出的。1975 年至 1979 年，IBM 公司 San-jose Research Laboratory 研制的关系数据库管理系统原型系统 System R 实现了这种语言。由于它功能丰富，语言简洁，使用方法灵活，而备受用户和计算机业界的青睐，被众多计算机公司和软件公司采用。经过多年的发展，SQL 语言已成为关系数据库的标准语言。

1.3.2　SQL 语言的组成

SQL 语言主要由以下几部分组成：

- **数据定义语言**（Data Definition Language，DDL）；
- **数据操纵语言**（Data Manipulation Language，DML）；
- **数据控制语言**（Data Control Language，DCL）；
- **其他语言要素**（Additional Language Elements）。

SQL 语句的数目、种类较多，其主体大约由 40 条语句组成，详细信息可参考相关资料，在此不再作一一解释。

1.3.3　SQL 语句的结构

所有的 SQL 语句均有自己的结构，如图 1-2 所示。每条 SQL 语句均由一个谓词开头，该谓词描述这条语句要产生的动作，如 Select。谓词后紧接着一个或多个子句（Clause），子句中给出了被谓词作用的数据或提供动作的详细信息。每一条子句由一个关键字开始，如 WHERE。

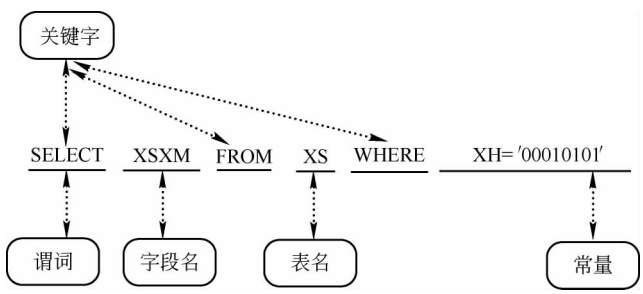


图 1-2　SQL 语句基本结构图

1.3.4　常用的 SQL 语句

在使用数据库时用得最多的是数据库操纵语言（DML，Data Manipulation Language）。DML 包含了最常用的核心 SQL 语句，即 SELECT、INSERT、UPDATE、DELETE。下面将对这些核心 SQL 语句一一作简单介绍。

1. 数据查询语句

在数据库中，数据查询是通过 SELECT 语句来完成的。SELECT 语句可以从数据库中按用户要求检索数据，并将结果以二维表的形式返回。

简单的 SELECT 语句的语法如下：

```
SELECT [ALL|DINSTINCT]<目标表达式> [,<目标表达式>]
FROM <表或视图名> [,<表或视图名>]
[WHERE<条件表达式>]
```

[GROUP BY<列名 1>][HAVING<条件表达式>]
[ORDER BY<列名 2>][ASC|DESC]

整个 SQL 语句的含义是：根据 WHERE 子句的条件表达式从 FROM 子句指定的基本表或视图中找出满足条件的元组，再按 SELECT 子句中的目标列表达式选出元组中的属性值形成结果表。如果有 GROUP 子句，则将结果按<列名 1>的值进行分组，该属性列值相等的元组为一个组，每个组产生结果表中的一条记录。如果 GROUP 子句带有 HAVING 短语，则只有满足指定条件的组才予以输出。如果有 ORDER 子句，则结果表还要按<列名 2>的值升序或降序排序。下面对 SEECT 语句的常用形式举例说明。

本章中所有例子的运行结果都可以通过 SQL Server 查询分析器获得。可以将每个例子给出的 SQL 语句自行键入到 SQL Server 查询分析器的 SQL 编辑窗口中，然后执行，即可获得各例的查询结果。

例 1-5　查询学生表 xs 中所有学生的学号、姓名、性别、生日。

```
USE db_xszhxxglxt
SELECT xh,xsxm,xsxb,xssr
FROM xs
```

USE db_xszhxxglxt 语句表示其后紧跟的 SQL 语句中使用的数据库表 xs 所在的数据库是 db_xszhxxglxt。如果在打开的 SQL Server 查询分析器窗体中已经选择了对应的数据库，则不需要在 SQL 查询语句前加上 USE 语句了。如果没有使用以上任何一种方式指定对应的数据库，那么 SQL Server 查询分析器在执行 SQL 语句时会报错。

运行结果如图 1-3 所示。



	xh	xsxm	xsxb	xssr
1	00010101	邹凯	男	1980-09-12
2	00010102	花宇	女	1981-07-19
3	00010103	刘军	男	1982-01-28
4	00010104	潘晓明	男	1980-10-03
5	00010105	刘安	男	1982-09-01
6	00010201	张三	男	1980-01-23
7	00010202	李四	男	1981-09-13
8	00010203	王明	男	1980-12-01
9	00010204	王五	男	1982-09-01
10	00010205	李强	男	1981-11-04
11	00020101	李小倩	女	1982-11-21
12	00020102	戴晓明	男	1983-10-08

图 1-3　查询结果界面

需要全本请访问 www.ertongbook.com