



“十二五”职业教育国家规划教材
经全国职业教育教材审定委员会审定

高等职业院校教学改革创新示范教材 计算机系列规划教材

SQL Server 实用教程（第4版）

◎ 刘启芬 编著



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>



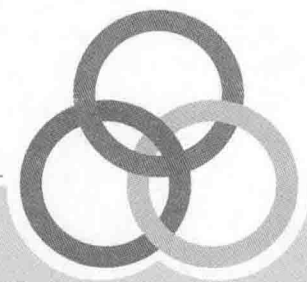
“十二五”职业教育国家规划教材
经全国职业教育教材审定委员会审定

高等职业院校教学改革创新示范教材·计算机系列规划教材

SQL Server 实用教程

第4版

刘启芬 编著



電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书以Microsoft SQL Server 2008中文版为平台,系统介绍数据库的基本概念、数据库创建、表与表数据操作、数据库的查询和视图、T-SQL语言、索引与数据完整性、存储过程和触发器、备份与恢复、系统安全管理等,界面操作和命令同步介绍,通过实例掌握命令主要参数。学生对照实例进行实验,消化介绍内容。最后是综合应用实习,包括当前市场最流行的PHP、JavaEE、ASP.NET和Visual Basic操作SQL Server 2008数据库的基本功能,学生熟悉其一可了解其他,方便教学与应用。

本书免费提供教学课件和配套的客户端/SQL Server 2008应用系统数据库和所有源程序文件。

本书可作为高职高专数据库课程教材和社会培训教材,也可供广大数据库应用开发人员参考。

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。

图书在版编目(CIP)数据

SQL Server 实用教程 / 刘启芬编著. —4版. —北京: 电子工业出版社, 2014.8

“十二五”职业教育国家规划教材

ISBN 978-7-121-23925-0

I. ①S… II. ①刘… III. ①关系数据库系统—高等职业教育—教材 IV. ①TP311.138

中国版本图书馆 CIP 数据核字 (2014) 第 172857 号

策划编辑: 程超群

责任编辑: 郝黎明

印 刷: 三河市鑫金马印装有限公司

装 订: 三河市鑫金马印装有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1 092 1/16 印张: 19.25 字数: 499 千字

版 次: 2002 年 8 月第 1 版

2014 年 8 月第 4 版

印 次: 2014 年 8 月第 1 次印刷

定 价: 39.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

PREFACE



Microsoft SQL Server 是目前最流行的大中型关系型数据库管理系统 (DBMS)，2002 年，我们结合教学和应用开发实践，编写了《SQL Server 实用教程》，该教材受到高校师生和广大读者的广泛好评，两年多重印 10 次。《SQL Server 实用教程 (第 2 版)》对 SQL Server 基本内容进行了优化改进，并在客户端编程实习方面统一了功能要求，又获得了很大成功，三年内重印 11 次。《SQL Server 实用教程 (第 3 版)》将 SQL Server 2000 升级到 SQL Server 2005，对 SQL Server 2005 进行了系统介绍，同时对实习模式进行了规范化。

高职版《SQL Server 实用教程 (第 4 版)》(“十二五”职业教育国家规划教材)，根据高等职业教育的特点，以应用为目的，以方便教和学为宗旨而编写。

本书以 Microsoft SQL Server 2008 中文版为平台，系统介绍数据库的基本概念、数据库创建、表与表数据操作、数据库的查询和视图、T-SQL 语言、索引与数据完整性、存储过程和触发器、备份与恢复、系统安全管理等。界面操作和命令同步介绍，通过实例掌握命令主要参数。学生对照实例进行实验，消化介绍内容。最后是综合应用实习，所有实习操作相同数据库，实习 0 创建该数据库，综合应用操作数据库的主要对象和主要命令。客户端开发平台包括当前市场最流行的 PHP、JavaEE、ASP.NET 和 Visual Basic 操作 SQL Server 2008 数据库的基本功能，学生熟悉其一可了解其他，方便教学与应用，以方便读者学习 SQL Server 2008 和应用它解决问题。

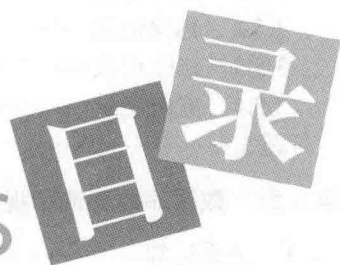
本书由南京师范大学刘启芬老师编著。参加本书编写的还有丁有和、徐文胜、殷红先、曹弋、陈瀚、陈冬霞、邓拼搏、高茜、刘博宇、彭作民、钱晓军、孙德荣、陶卫冬、吴明祥、王志瑞、徐斌、俞琰、严大牛、郑进、张为民、周何骏、周怡君、于金彬、马骏、周怡明、顾韵华、姜乃松、梁敬东等。此外，还有许多同志对本书的编写提供了很多帮助，在此一并表示感谢！

由于作者水平有限，不当之处在所难免，恳请读者批评指正。本书提供丰富的教学课件和配套的客户端/SQL Server 2008 应用系统数据库以及所有源程序文件，可从华信教育资源网 (www.hxedu.com.cn) 免费下载。本书附录 B、C、D，请读者扫描二维码进行查阅。

意见建议邮箱：easybooks@163.com

编 者

CONTENTS



第 1 章 数据库的基本概念	1
1.1 数据库基本概念	1
1.1.1 数据库与数据库管理系统	1
1.1.2 数据模型	2
1.1.3 关系型数据库语言	4
1.2 数据库设计	4
1.2.1 概念结构设计	4
1.2.2 逻辑结构设计	5
1.2.3 数据库物理设计	7
1.3 数据库应用系统	7
1.3.1 数据库的连接方式	7
1.3.2 客户-服务器 (C/S) 模式 应用系统	9
1.3.3 三层客户-服务器 (B/S) 模式 应用系统	10
1.4 SQL Server 2008 环境	11
1.4.1 SQL Server 2008 概述	11
1.4.2 SQL Server 2008 的安装	11
1.4.3 SQL Server 2008 服务器 组件	18
1.4.4 SQL Server 2008 管理和 开发工具	19
习题 1	23
第 2 章 数据库创建	24
2.1 SQL Server 数据库基本概念	24
2.1.1 逻辑数据库	24
2.1.2 物理数据库	26

2.1.3 系统数据库和用户数据库	27
2.2 界面方式创建数据库	27
2.2.1 数据库的创建	27
2.2.2 数据库的修改和删除	29
2.3 命令方式创建数据库	32
2.3.1 创建数据库	33
2.3.2 修改数据库	36
2.3.3 删除数据库	38
2.3.4 数据库快照	38
习题 2	39
第 3 章 表与表数据操作	40
3.1 表结构和数据类型	40
3.1.1 表和表结构	40
3.1.2 数据类型	41
3.1.3 表结构设计	46
3.2 界面方式操作表	47
3.2.1 创建表	47
3.2.2 修改表结构	49
3.2.3 删除表	52
3.3 命令方式操作表	52
3.3.1 创建表	52
3.3.2 创建分区表	53
3.3.3 修改表结构	57
3.3.4 删除表	58
3.4 界面方式操作表数据	58
3.4.1 插入记录	59
3.4.2 删除记录	59
3.4.3 修改记录	60

3.5 命令方式操作表数据.....	60	5.2.2 用户自定义数据类型.....	110
3.5.1 插入记录.....	60	5.2.3 变量.....	113
3.5.2 删除记录.....	63	5.3 运算符与表达式.....	117
3.5.3 修改记录.....	64	5.4 流程控制语句.....	122
3.5.4 更新记录.....	65	5.4.1 BEGIN...END 语句块.....	122
习题3.....	66	5.4.2 条件语句.....	122
第4章 数据库的查询和视图.....	68	5.4.3 CASE 语句.....	124
4.1 关系运算.....	68	5.4.4 无条件转移语句.....	125
4.2 数据库的查询.....	70	5.4.5 循环语句.....	125
4.2.1 <select_list>: 选择输出列.....	71	5.4.6 返回语句.....	126
4.2.2 WHERE 子句: 指定查询 条件.....	75	5.4.7 等待语句.....	127
4.2.3 FROM 子句: 指定查询 对象.....	80	5.4.8 错误处理语句.....	127
4.2.4 连接: 实现多表查询.....	82	5.5 系统内置函数.....	128
4.2.5 GROUP BY 子句: 对查询 结果分组.....	85	5.5.1 系统内置函数介绍.....	128
4.2.6 HAVING 子句: 对分组数据 进行筛选.....	89	5.5.2 常用系统标量函数.....	129
4.2.7 ORDER BY 子句: 查询结果 排序.....	90	5.6 用户定义函数.....	135
4.2.8 SELECT 语句的其他语法.....	92	5.6.1 用户函数的定义与调用.....	135
4.3 视图.....	94	5.6.2 用户定义函数的删除.....	140
4.3.1 视图概念.....	94	习题5.....	140
4.3.2 创建视图.....	95	第6章 索引与数据完整性.....	141
4.3.3 查询视图.....	98	6.1 索引.....	141
4.3.4 更新视图.....	99	6.1.1 索引的分类.....	141
4.3.5 修改视图的定义.....	101	6.1.2 索引的创建.....	142
4.3.6 删除视图.....	102	6.1.3 重建索引.....	145
4.4 游标.....	102	6.1.4 索引的删除.....	145
4.4.1 游标概念.....	102	6.2 数据完整性.....	146
4.4.2 声明游标.....	103	6.2.1 数据完整性的分类.....	146
4.4.3 打开游标.....	103	6.2.2 实体完整性的实现.....	148
4.4.4 读取游标数据.....	104	6.2.3 域完整性的实现.....	150
4.4.5 关闭游标和删除游标.....	105	6.2.4 参照完整性的实现.....	155
习题4.....	105	习题6.....	158
第5章 T-SQL 语言.....	107	第7章 存储过程和触发器.....	159
5.1 SQL 语言与 T-SQL 语言.....	107	7.1 存储过程.....	159
5.2 常量、变量与数据类型.....	108	7.1.1 存储过程的类型.....	159
5.2.1 常量.....	108	7.1.2 存储过程的创建与执行.....	160
		7.1.3 存储过程的修改.....	163
		7.1.4 存储过程的删除.....	164
		7.1.5 界面方式操作存储过程.....	164
		7.2 触发器.....	166

7.2.1 触发器的类型	166	9.4.3 撤销权限	218
7.2.2 触发器的创建	166	9.5 数据库架构的定义和使用	219
7.2.3 触发器的修改	172	9.5.1 界面方式创建架构	219
7.2.4 触发器的删除	173	9.5.2 命令方式创建架构	222
7.2.5 界面方式操作触发器	173	习题 9	222
习题 7	174	第 10 章 其他概念	223
第 8 章 备份与恢复	175	10.1 事务	223
8.1 备份和恢复概述	175	10.1.1 事务与 ACID 属性	223
8.1.1 备份和恢复需求分析	175	10.1.2 多用户使用的问题	224
8.1.2 数据库备份的基本概念	175	10.1.3 事务处理	224
8.1.3 数据库恢复概念	177	10.1.4 事务隔离级	227
8.2 备份操作和备份命令	178	10.2 锁定	229
8.2.1 创建备份设备	178	10.2.1 锁定粒度	229
8.2.2 备份命令	180	10.2.2 锁定模式	229
8.2.3 界面方式进行备份	184	10.3 自动化管理	230
8.3 恢复操作和恢复命令	186	10.3.1 SQL Server 代理	231
8.3.1 检查点	186	10.3.2 操作员	232
8.3.2 数据库的恢复命令	186	10.3.3 作业	233
8.3.3 图形向导方式恢复数据库	191	10.3.4 警报	237
8.4 复制数据库	193	习题 10	240
8.5 附加数据库	195	实习 0 创建实习数据库	241
习题 8	196	P0.1 创建数据库及其对象	241
第 9 章 系统安全管理	197	P0.2 功能和界面	243
9.1 SQL Server 2008 的安全机制	197	实习 1 PHP/SQL Server 2008 学生成绩管理系统	246
9.1.1 SQL Server 2008 的身份验证 模式	197	P1.1 PHP 开发平台搭建	246
9.1.2 SQL Server 2008 的安全性 机制	198	P1.1.1 安装 PHP 环境	246
9.2 建立和管理用户账户	199	P1.1.2 Eclipse 安装与配置	248
9.2.1 界面方式管理用户账户	199	P1.2 创建 PHP 项目	249
9.2.2 命令方式管理用户账户	202	P1.2.1 PHP 项目的建立	249
9.3 角色管理	204	P1.2.2 PHP 连接 SQL Server	251
9.3.1 固定服务器角色	204	P1.3 系统登录及界面设计	252
9.3.2 固定数据库角色	206	P1.4 学生功能	253
9.3.3 自定义数据库角色	209	P1.4.1 修改口令	253
9.3.4 应用程序角色	211	P1.4.2 查询成绩	255
9.4 数据库权限的管理	214	P1.5 教师功能	255
9.4.1 授予权限	214	P1.5.1 增减学生	255
9.4.2 拒绝权限	217	P1.5.2 输入成绩	257

实习 2 JavaEE/SQL Server 2008

学生成绩管理系统 260

P2.1 JavaEE 开发平台搭建 260

P2.1.1 安装软件 260

P2.1.2 环境整合 260

P2.2 创建 Struts 2 项目 262

P2.2.1 加载 Struts 2 包 262

P2.2.2 连接 SQL Server 263

P2.3 系统登录及界面设计 266

P2.3.1 界面设计 266

P2.3.2 登录功能 267

P2.4 学生功能 269

P2.4.1 修改口令 269

P2.4.2 查询成绩 272

实习 3 ASP.NET4/SQL Server 2008

学生成绩管理系统 274

P3.1 ADO.NET 架构原理 274

P3.2 创建 ASP.NET 项目 275

P3.2.1 ASP.NET 项目的建立 275

P3.2.2 ASP.NET 连接 SQL Server 275

P3.3 系统登录及界面设计 276

P3.4 学生功能 279

P3.4.1 修改口令 279

P3.4.2 查询成绩 281

P3.5 教师功能 282

P3.5.1 增减学生 282

P3.5.2 输入成绩 287

实习 4 VB6.0/SQL Server 2008

学生成绩管理系统 288

P4.1 VB 数据库开发准备 288

P4.1.1 创建 ODBC 数据源 288

P4.1.2 新建 VB 6.0 项目 289

P4.1.3 连接数据库 290

P4.2 系统登录功能 290

P4.3 学生功能 291

P4.3.1 修改口令 291

P4.3.2 查询成绩 292

P4.4 教师功能 294

P4.4.1 增减学生 294

P4.4.2 输入成绩 295

附录 A 学生成绩 (PXSCJ) 数据库

表样本数据 296

附录 B 常用语句、全局变量和常用函数



附录 C SQL Server 2008 导入/导出 Word 等类型文件——采用 ASP.NET(C#)实现



附录 D CLR 存储过程和触发器



数据库的基本概念

为了更好地学习 SQL Server，首先需要介绍一下数据库的基本概念，如果学习过数据库原理，那么本章数据库原理部分仅仅作为一个参考。

1.1 数据库基本概念

1.1.1 数据库与数据库管理系统

1. 数据库

数据库 (DB) 是存放数据的仓库，只不过这些数据存在一定的关联，并按一定的格式存放在计算机内。广义上讲，数据不仅包含数字，还包括文本、图像、音频、视频等。

例如，把一个学校的学生、课程、学生成绩等数据有序地组织并存放在计算机内，就可以构成一个数据库。因此，数据库由一些持久的相互关联的数据的集合组成，并以一定的组织形式存放在计算机的存储介质中。

2. 数据库管理系统

数据库管理系统 (DBMS) 是管理数据库的系统，它按一定的数据模型组织数据。DBMS 应提供如下功能。

- (1) 数据定义功能：可定义数据库中的数据对象。
- (2) 数据操纵功能：可对数据库表进行基本操作，如插入、删除、修改、查询等。
- (3) 数据的完整性检查功能：保证用户输入的数据满足相应的约束条件。
- (4) 数据库的安全保护功能：保证只有具有权限的用户才能访问数据库中的数据。
- (5) 数据库的并发控制功能：使多个应用程序可在同一时刻并发地访问数据库的数据。
- (6) 数据库系统的故障恢复功能：使数据库在运行出现故障时进行数据库恢复，以保证数据库可靠运行。
- (7) 在网络环境下访问数据库的功能。
- (8) 方便、有效地存取数据库信息的接口和工具。编程人员通过程序开发工具与数据库的接口编写数据库应用程序。数据库系统管理员 (DBA, DataBase Administrator) 通过提供的工具对数据库进行管理。

数据、数据库、数据库管理系统与操作数据库的应用程序，加上支撑它们的硬件平台、软件平台和与数据库有关的人员一起构成了一个完整的数据库系统。图 1.1 描述了数据库系统的构成。

自 20 世纪 70 年代关系模型提出后，商用数据库系统迅速采用了这种模型，涌现出很多性能优良的关系数据库管理系统 (RDBMS)。

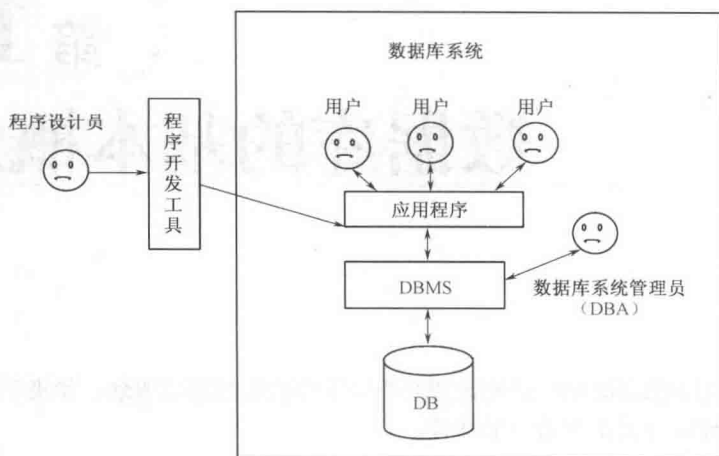


图 1.1 数据库系统的构成

目前，商品化的数据库管理系统以关系型数据库为主导产品，技术比较成熟。主流的关系型数据库管理系统包括 Oracle、SQL Server、DB2、Sybase、Informix 和 Ingers 等，小型的关系型数据库管理系统包括 MySQL、Access、Visual FoxPro 等。

SQL Server 是由 Microsoft 公司开发和推广的在 Windows 平台上最为流行的大中型关系数据库管理系统。SQL Server 版本从 SQL Server 6.5、7.0、2000 到 2005，再到目前的 SQL Server 2008，功能不断完善。本书主要介绍 SQL Server 2008。

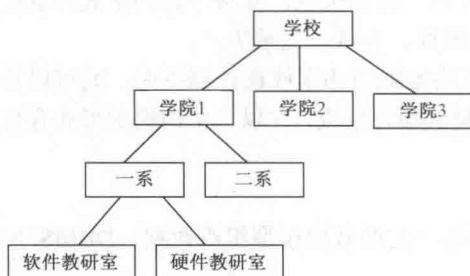


图 1.2 某学校按层次模型组织的数据示例

1.1.2 数据模型

数据库管理系统根据数据模型对数据进行存储和管理，数据库管理系统采用的数据模型主要有：层次模型、网状模型和关系模型。

(1) 层次模型：以树状层次结构组织数据。图 1.2 所示为某学校按层次模型组织的数据示例。

(2) 网状模型：每一个数据用一个节点表示，每个节点与其他节点都有联系，这样，数据库中的所有数据节点就构成了一个复杂的网络。图 1.3 所示为按网状模型组织的数据示例。

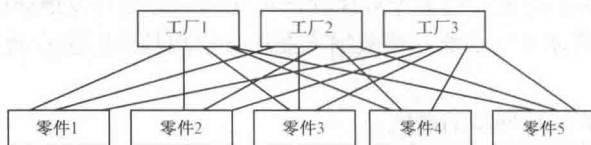


图 1.3 按网状模型组织的数据示例

(3) 关系模型：以二维表格（关系表）的形式组织数据库中的数据。

例如，在学生成绩管理系统所涉及的学生、课程和成绩三个表中。学生表涉及的主要信息有学号、姓名、性别、出生时间、专业、总学分、备注。课程表涉及的主要信息有课程号、课程名、开课学期、学时和学分。成绩表涉及的主要信息有学号、课程号和成绩。表 1.1、表 1.2 和表 1.3 分别描述了学生成绩管理系统中学生、课程和成绩这三个表的部分数据。

表 1.1 学生表

学 号	姓 名	性 别	出 生 时 间	专 业	总 学 分	备 注
081101	王林	男	1990-02-10	计算机	50	
081103	王燕	女	1989-10-06	计算机	50	
081108	林一帆	男	1989-08-05	计算机	52	已提前修完一门课
081202	王林	男	1989-01-29	通信工程	40	有一门课不及格, 待补考
081204	马琳琳	女	1989-02-10	通信工程	42	

表 1.2 课程表

课 程 号	课 程 名	开 课 学 期	学 时	学 分
0101	计算机基础	1	80	5
0102	程序设计与语言	2	68	4
0206	离散数学	4	68	4

表 1.3 成绩表

学 号	课 程 号	成 绩	学 号	课 程 号	成 绩
081101	101	80	081108	101	85
081101	102	78	081108	102	64
081101	206	76	081108	206	87
081103	101	62	081202	101	65
081103	102	70	081204	101	91

表格中的一行称为一个记录, 一列称为一个字段, 每列的标题称为字段名。如果给每个关系表取一个名字, 则有 n 个字段的关系表的结构可表示为关系表名 (字段名 1, ..., 字段名 n), 通常把关系表的结构称为关系模式。

在关系表中, 如果一个字段或几个字段组合的值可唯一标识其对应记录, 则称该字段或字段组合为码。

例如, 表 1.1 中的“学号”可唯一标识每一个学生, 表 1.2 中的“课程号”可唯一标识每一门课。表 1.3 中的“学号”和“课程号”可唯一标识每一个学生一门课程的成绩。

有时, 一个表可能有多个码, 比如表 1.1 中, 姓名不允许重名, 则“学号”、“姓名”均是学生信息表码。对于每一个关系表, 通常可指定一个码为“主码”, 在关系模式中, 一般用下划线标出主码。

设表 1.1 的名字为 XSB, 关系模式可分别表示为 XSB (学号, 姓名, 性别, 出生时间, 专业, 总学分, 备注)。

设表 1.2 的名字为 KCB, 关系模式可分别表示为 KCB (课程号, 课程名, 开课学期, 学时, 学分)。

设表 1.3 的名字为 CJB, 关系模式可分别表示为 CJB (学号, 课程号, 成绩, 学分)。

从上面可看出, 按关系模型组织的数据表达方式简洁、直观, 插入、删除、修改操作方便, 而按层次、网状模型组织的数据表达方式复杂, 插入、删除、修改操作也很复杂。因此, 关系模型得到了广泛应用, SQL Server 2008 是支持关系数据模型的数据库管理系统。

1.1.3 关系型数据库语言

关系型数据库的标准语言是 SQL (Structured Query Language, 结构化查询语言)。SQL 语言是用于关系数据库查询的结构化语言, 最早由 Boyce 和 Cha MBedin 在 1974 年提出, 称为 SEQUEL 语言。1976 年, IBM 公司的 San Jose 研究所在研制关系数据库管理系统 System R 时修改为 SEQUEL2, 即目前的 SQL 语言。

1986 年 10 月, 美国 ANSI 对 SQL 进行规范后, 以此作为关系数据库管理系统的标准语言。

作为关系数据库的标准语言, 它已被众多商用数据库管理系统产品所采用, 不过, 不同的数据库管理系统在其实践过程中都对 SQL 规范做了某些改变和扩充。所以, 实际上, 不同数据库管理系统之间的 SQL 语言不能完全通用。例如, 微软公司的 MS SQL-Server 支持的是 T-SQL, 而甲骨文公司的 Oracle 数据库所使用的 SQL 语言则是 PL-SQL。

1.2 数据库设计

1.2.1 概念结构设计

通常, 把每一类数据对象的个体称为“实体”, 而每一类对象个体的集合称为“实体集”, 因此, 在学生成绩管理系统中主要涉及“学生”和“课程”两个实体集。

其他非主要的实体可以很多, 如班级、班长、任课教师、辅导员等实体。把每个实体集涉及的信息项称为属性。就“学生”实体集而言, 它的属性有: 学号、姓名、性别、出生时间、专业、总学分和备注。“课程”实体集属性有: 课程号、课程名、开课学期、学时和学分。

实体集中的实体彼此是可区别的。如果实体集中的属性或最小属性组合的值能唯一标识其对应实体, 则将该属性或属性组合称为码。对于每一个实体集, 可指定一个码为主码。

如果用矩形框表示实体集, 用带半圆的矩形框表示属性, 用线段连接实体集与属性, 当一个属性或属性组合指定为主码时, 在实体集与属性的连接线上标记一斜线, 则可以用图 1.4 描述学生成绩管理系统中的实体集及每个实体集涉及的属性。

实体集 A 和实体集 B 之间存在各种关系, 通常把这些关系称为“联系”。通常将实体集及实体集联系的图表示称为实体 (Entity) - 联系 (Relationship) 模型。

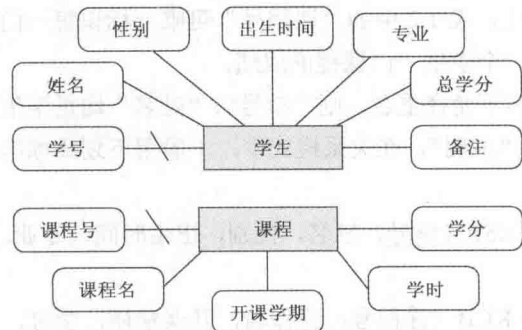


图 1.4 学生和课程实体集属性的描述

E-R 图就是 E-R 模型的描述方法, 即实体-联系图, 通常, 关系数据库的设计者使用 E-R 图来对信息世界建模。在 E-R 图中, 使用矩形表示实体型, 使用椭圆表示属性, 使用菱形表示联系。从分析用户项目涉及的数据对象及数据对象之间的联系出发, 到获取 E-R 图的这一过程, 称为概念结构设计。

两个实体集 A 和 B 之间的联系可能是以下三种情况之一。

1. 一对一的联系 (1:1)

A 中的一个实体至多与 B 中的一个实体相联系, B 中的一个实体也至多与 A 中的一个实体相联系。例如,“班级”与“正班长”这两个实体集之间的联系是一对一的联系,因为一个班级只有一个正班长,反过来,一个正班长只属于一个班级。“班级”与“正班长”两个实体集的 E-R 模型如图 1.5 所示。

2. 一对多的联系 (1:n)

A 中的一个实体可以与 B 中的多个实体相联系,而 B 中的一个实体至多与 A 中的一个实体相联系。例如,“班级”与“学生”这两个实体集之间的联系是一对多的联系,因为,一个班级可有若干学生,反过来,一个学生只能属于一个班级。“班级”与“学生”两个实体集的 E-R 模型如图 1.6 所示。

3. 多对多的联系 (m:n)

A 中的一个实体可以与 B 中的多个实体相联系,而 B 中的一个实体也可与 A 中的多个实体相联系。例如,“学生”与“课程”这两个实体集之间的联系是多对多的联系,因为,一个学生可选多门课程,反过来,一门课程可被多个学生选修。“学生”与“课程”两个实体集的 E-R 模型如图 1.7 所示。

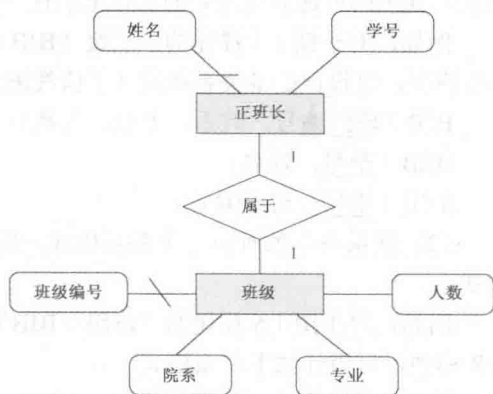


图 1.5 “班级”与“正班长”实体集 E-R 模型

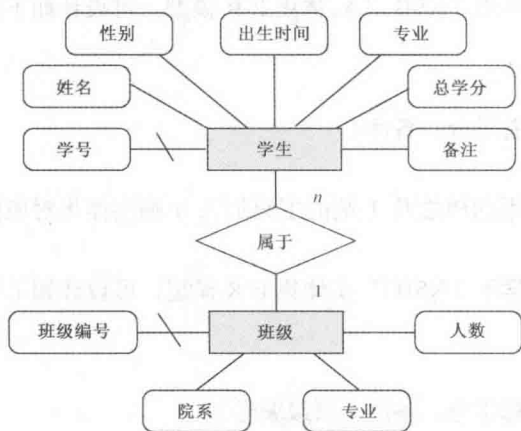


图 1.6 “学生”与“班级”两个实体集的 E-R 模型

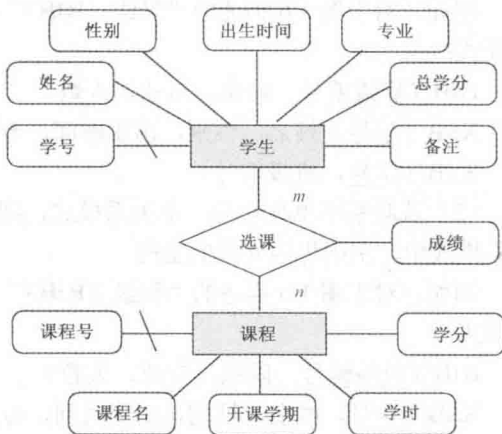


图 1.7 “学生”与“课程”实体集间的 E-R 模型

1.2.2 逻辑结构设计

用 E-R 图描述学生成绩管理系统中实体集与实体集之间的联系,目的是以 E-R 图为工具,设计关系型的数据库,即确定应用系统所使用的数据库应包含哪些表,每个表的结构是怎样的。前面已介绍了实体集之间的联系,下面将介绍根据联系从 E-R 图得到关系模式的方法。

1. 1:1 联系的 E-R 图到关系模式的转换

1:1 的联系既可单独对应一个关系模式,也可以不单独对应一个关系模式。

(1) 联系单独对应一个关系模式,则由联系属性、参与联系的各实体集的主码属性构成关系

模式，其主码可选参与联系的实体集的任一方的主码。

例如，对于图 1.5 描述的“班级 (BJB)”与“正班长 (BZB)”实体集通过属于 (SYB) 联系 E-R 模型，可设计如下关系模式（下横线表示该字段为主码）：

BJB (班级编号，院系，专业，人数)

BZB (学号，姓名)

SYB (学号，班级编号)

(2) 联系不单独对应一个关系模式，联系的属性及一方的主码加入另一方实体集对应的关系模式中。

例如，对于图 1.5 描述的“班级 (BJB)”与“正班长 (BZB)”实体集通过属于 (SYB) 联系 E-R 模型，可设计如下关系模式：

BJB (班级编号，院系，专业，人数)

BZB (学号，姓名，班级编号)

或者

BJB (班级编号，院系，专业，人数，学号)

BZB (学号，姓名)

2. 1:n 联系的 E-R 图到关系模式的转换

1:n 的联系既可单独对应一个关系模式，也可以不单独对应一个关系模式。

(1) 若联系单独对应一个关系模式，则由联系的属性、参与联系的各实体集的主码属性构成关系模式，n 端的主码作为该关系模式的主码。

例如，对于图 1.6 描述的“班级 (BJB)”与“学生 (XSB)”实体集 E-R 模型，可设计如下关系模式：

BJB (班级编号，院系，专业，人数)

XSB (学号，姓名，性别，出生时间，专业，总学分，备注)

SYB (学号，班级编号)

(2) 若联系不单独对应一个关系模式，则将联系的属性及 1 端的主码加入 n 端实体集对应的关系模式中，主码仍为 n 端的主码。

例如，对于图 1.6 描述的“班级 (BJB)”与“学生 (XSB)”实体集 E-R 模型，可设计如下关系模式：

BJB (班级编号，院系，专业，人数)

XSB (学号，姓名，性别，出生时间，专业，总学分，备注，班级编号)

3. m:n 联系的 E-R 图到关系模式的转换

m:n 的联系单独对应一个关系模式，该关系模式包括联系的属性、参与联系的各实体集的主码属性，该关系模式的主码由各实体集的主码属性共同组成。

例如，对于图 1.7 描述的“学生 (XSB)”与“课程 (KCB)”实体集之间的联系可设计如下关系模式：

XSB (学号，姓名，性别，出生时间，专业，总学分，备注)

KCB (课程号，课程名称，开课学期，学时，学分)

CJB (学号，课程号，成绩)

关系模式 CJB 的主码是由“学号”和“课程号”两个属性组合起来构成的一个主码，一个关系模式只能有一个主码。

至此，已介绍了根据 E-R 图设计关系模式的方法，通常这一设计过程称为逻辑结构设计。

在设计好一个项目的关系模式后，就可以在数据库管理系统环境下，创建数据库、关系表及其他数据库对象，输入相应数据，并根据需要对数据库中的数据进行各种操作。

1.2.3 数据库物理设计

数据的物理模型指数据的存储结构，如对数据库物理文件、索引文件的组织方式、文件的存取路径，内存的管理等。物理模型对用户是不可见的，它不仅与数据库管理系统有关，还与操作系统甚至硬件有关。

1.3 数据库应用系统

1.3.1 数据库的连接方式

客户端应用程序或应用服务器向数据库服务器请求服务时，必须首先和数据库建立连接。虽然不同的 RDBMS 都遵循 SQL 标准，但不同厂家开发的 RDBMS 有差异，如存在适应性和可移植性等方面的问题。因此，人们开始研究和开发连接不同 RDBMS 的通用方法、技术和软件。

1. ODBC 数据库接口

ODBC 即开放式数据库互连 (Open DataBase Connectivity)，是微软公司推出的一种实现应用程序和关系数据库之间通信的接口标准。符合标准的数据库就可以通过 SQL 语言编写的命令对数据库进行操作，但只针对关系数据库。目前所有的关系数据库都符合该标准 (如 SQL Server, Oracle, Access, Excel 等)。

ODBC 本质上是一组数据库访问 API (应用程序编程接口)，它由一组函数调用组成，核心是 SQL 语句，其结构如图 1.8 所示。



图 1.8 ODBC 数据库接口

在具体操作时，首先必须用 ODBC 管理器注册一个数据源。管理器根据数据源提供的数据库位置、数据库类型及 ODBC 驱动程序等信息，建立起 ODBC 与具体数据库的联系。这样，只要应用程序将数据源名提供给 ODBC，ODBC 就能建立起与相应数据库的连接。

2. OLE DB 数据库接口

OLE DB 即数据库链接和嵌入对象 (Object Linking and EMbedding DataBase)。OLE DB 是微软提出的基于 COM 思想且面向对象的一种技术标准，其目的是提供一种统一的数据访问接口来访问

问各种数据源。

这里所说的“数据”除了标准的关系型数据库中的数据之外，还包括邮件数据、Web 上的文本或图形、目录服务（Directory Services）、主机系统中的文件和地理数据以及自定义业务对象等。

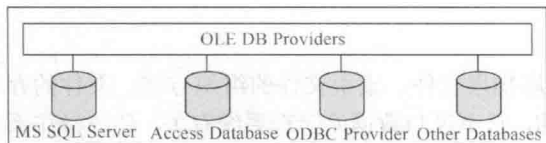


图 1.9 OLE DB 数据库接口

OLE DB 标准的核心内容就是，提供一种相同的访问接口，使得数据的使用者（应用程序）可以使用同样的方法访问各种数据，而不用考虑数据的具体存储地点、格式或类型，其结构如图 1.9 所示。

3. ADO 数据库接口

ADO（ActiveX Data Objects）是微软公司开发的基于 COM 的数据库应用程序接口，通过 ADO 连接数据库，可以灵活地操作数据库中的数据。

图 1.10 所示为应用程序通过 ADO 访问 SQL Server 数据库接口。从图中可看出，使用 ADO 访问 SQL Server 数据库有两种途径：一种是通过 ODBC 驱动程序，另一种是通过 SQL Server 专用的 OLE DB Provider，后者有更高的访问效率。

随着网络技术的发展，网络数据库以及相关的操作技术也越来越多地应用到实际中，而数据库操作技术也在不断地发展完善。ADO 对象模型进一步发展成了 ADO.NET。ADO.NET 是 .NET Framework SDK 中用于操作数据库的类库总称。ADO.NET 相对于 ADO 的最大优势在于，对数据的更新修改可在与数据源完全断开连接的情况下进行，然后再把数据更新的结果和状态传回到数据源，这大大减少了过多的连接对数据库服务器资源的占用。

4. ADO.NET 数据库接口

ASP.NET 使用 ADO.NET 数据模型。该模型从 ADO 发展而来，但它不只是对 ADO 的改进，而是采用了一种全新的技术。主要表现在以下几个方面。

（1）ADO.NET 不是采用 ActiveX 技术，而是与 .NET 框架紧密结合的产物。

（2）ADO.NET 包含对 XML 标准的完全支持，这对于跨平台交换数据具有重要的意义。

（3）ADO.NET 既能在与数据源连接的环境下工作，又能在断开与数据源连接的条件下工作。特别是后者，非常适合于网络应用的需要。因为在网络环境下，保持与数据源连接，不符合网站的要求，不仅效率低，付出的代价高，而且常常会引发多个用户同时访问带来的冲突。因此 ADO.NET 系统集中主要精力来解决在断开与数据源连接的条件下的数据处理问题。

ADO.NET 提供了面向对象的数据库视图，并且在 ADO.NET 对象中封装了许多数据库属性和关系。最重要的是，ADO.NET 通过多种方式封装和隐藏了很多数据库访问的细节。用户可以完全不知道对象在与 ADO.NET 对象交互，也不用担心数据移动到另外一个数据库或者从另一个数据库获得数据的细节问题。图 1.11 所示为 ADO.NET 架构总览。

数据集是实现 ADO.NET 断开式连接的核心，从数据源读取的数据先缓存到数据集中，然后被程序或控件调用。这里的数据源可以是数据库或者 XML 数据。

数据提供者用于建立数据源与数据集之间的联系，它能连接各种类型的数据，并能按要求将数据源中的数据提供给数据集，或者从数据集向数据源返回处理后的数据。

5. JDBC 数据库接口

JDBC（Java DataBase Connectivity）是 Java Soft 公司开发的一组 Java 语言编写的用于数据库连接和操作的类和接口，可为多种 RDBMS 提供统一的访问方式。通过 JDBC 对数据库的访问包括 4 个主要组件：Java 应用程序、JDBC 驱动器管理器、驱动器和数据源。

在 JDBC API 中有两层接口：应用程序层和驱动程序层，前者使开发人员可以通过 SQL 调用数据库和取得结果，后者处理与具体数据库驱动程序相关的所有通信。

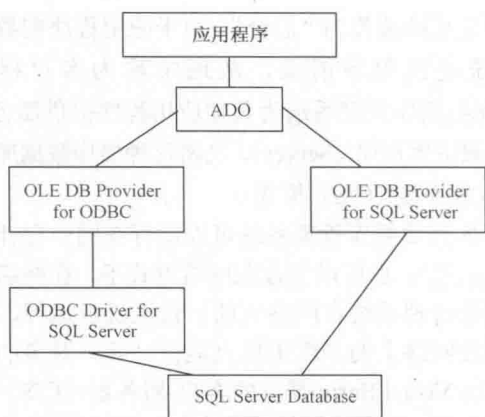


图 1.10 ADO 访问 SQL Server 的接口

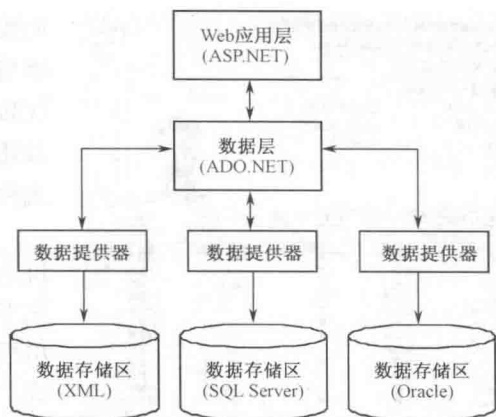


图 1.11 通过 ADO.NET 访问数据库的接口模型

使用 JDBC 接口对数据库操作有如下优点：

- (1) JDBC API 与 ODBC 十分相似，有利于用户理解。
- (2) 使编程人员从复杂的驱动器调用命令和函数中解脱出来，而致力于应用程序功能的实现。
- (3) JDBC 支持不同的关系数据库，增强了程序的可移植性。

使用 JDBC 的主要缺点：访问数据记录的速度会受到一定影响；此外，JDBC 结构中包含了不同厂家的产品，这给数据源的更改带来了较大麻烦。

6. 数据库连接池技术

网络环境下的数据库应用，由于用户众多，使用传统的 JDBC 方式进行数据库连接，系统资源开销过大，这成为制约大型企业级应用效率的瓶颈，而采用数据库连接池技术对数据库连接进行管理，可以大大提高系统的效率和稳定性。

1.3.2 客户-服务器 (C/S) 模式应用系统

Microsoft 公司开发的 SQL Server 数据库管理系统是当前最流行的数据库管理系统。数据库管理系统通过命令和适合专业人员的界面来操作数据库。

对于 SQL Server 2008 数据库管理系统，用户只需要在 SQL Server 2008 的 SQL Server Management Studio 管理工具中输入 SQL 命令，系统执行的结果就会返回到该工具上并显示出来。用户还可以直接通过该工具以界面方式操作数据库。

对于一般的数据库应用系统，除了数据库管理系统外，需要设计适合普通人员操作数据库的界面。目前，流行的开发数据库界面的工具主要包括 Visual BASIC、Visual C++、Visual FoxPro、Delphi、PowerBuilder 等。数据库应用程序与数据库、数据库管理系统之间的关系如图 1.12 所示。

从图中可看出，当应用程序需要处理数据库中的数据时，首先向数据库管理系统发送一个数据处理请求。数据库管理系统接收到这一请求后，对其进行分析，然后执行数据操作，并把操作结果返回给应用程序。

由于应用程序直接与用户打交道，而数据库管理系统不直接与用户打交道，所以应用程序被称为“前台”，

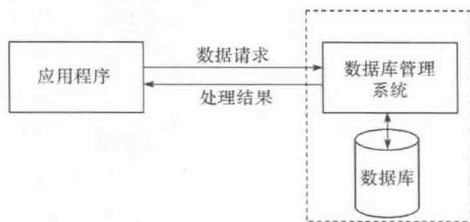


图 1.12 数据库应用程序与数据库、数据库管理系统之间的关系