

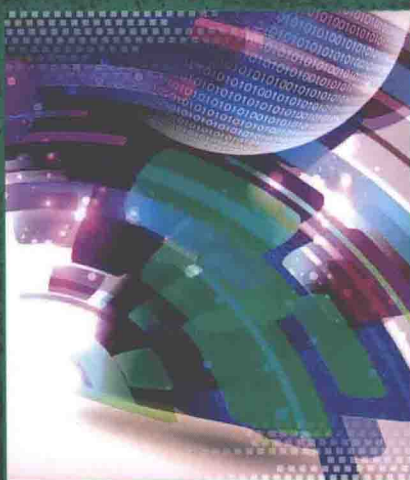


普通高等教育“十二五”精品规划教材

SQL Server 2005 数据库应用与开发

SQL Server 2005
SHUJUKU YINGYONG YU KAIFA

李新德 赵作辉 主 编



 北京理工大学出版社
BEIJING INSTITUTE OF TECHNOLOGY PRESS

普通高等教育“十二五”精品规划教材

SQL Server 2005 数据库应用与开发

主 编 李新德 赵作辉

副主编 蒋继东 任 莉

 北京理工大学出版社

BEIJING INSTITUTE OF TECHNOLOGY PRESS

图书在版编目 (CIP) 数据

SQL Server 2005 数据库应用与开发 / 李新德, 赵作辉主编. —北京:
北京理工大学出版社, 2012. 9 (2012. 10 重印)

ISBN 978 - 7 - 5640 - 5903 - 3

I. ①S… II. ①李… ②赵… III. ①关系数据库 - 数据库管理系统 - 高等学校 - 教材 IV. ①TP311. 138

中国版本图书馆 CIP 数据核字 (2012) 第 087522 号

出版发行 / 北京理工大学出版社

社 址 / 北京市海淀区中关村南大街 5 号

邮 编 / 100081

电 话 / (010)68914775(办公室) 68944990(批销中心) 68911084(读者服务部)

网 址 / <http://www.bitpress.com.cn>

经 销 / 全国各地新华书店

印 刷 / 北京慧美印刷有限公司

开 本 / 787 毫米 × 1092 毫米 1/16

印 张 / 16.5

字 数 / 382 千字

责任编辑 / 陈子慧

版 次 / 2012 年 9 月第 1 版 2012 年 10 月第 2 次印刷

陈莉华

印 数 / 2501 ~ 4000 册

责任校对 / 陈玉梅

定 价 / 43.00 元

责任印制 / 王美丽

前 言

SQL Server 2005 是由 Microsoft 公司于 2005 年 11 月推出的大型关系型数据库管理系统, 它很好地兼容了 SQL Server 2000 系统, 并在一些功能上得到了扩充和加强。SQL Server 2005 数据库管理系统被广泛应用在各行各业的生产、管理实践中, 具有良好的口碑和普及性, 各大院校都将其列为必修的课程。

本教材是作者在总结了多年的数据库应用程序开发和一线教学经验的基础上编撰而成的, 剔除了许多在实际应用中基本不用的内容。本教材以一个数据库应用程序“学生成绩管理系统”为例, 利用其数据全面、翔实地介绍了应用 SQL Server 2005 数据库管理系统进行数据库管理的各种操作以及数据库应用程序开发所需的各种知识和技能。通过对本教材的学习和具体的操作实践, 学生们可以快速地、全面地掌握 SQL Server 2005 数据库管理系统中常用的技术和技能, 为将来在实际工作中从事信息化相关技术工作打下坚实的基础。

本教材的编写具有以下特点:

(1) 以章、项目、任务为线索, 从大标题到具体任务层层细化, 内容的部署既保留了传统教材以章节为主线的编写模式, 便于知识的查询和检索, 又组织了以项目、任务为构造的实践操作模块, 适合于以操作为主的实际训练。

(2) 以实际应用为宗旨, 省略了许多生疏语法的内容, 代之以用界面操作来实现功能, 使得许多知识和技能简单易学, 避免了难学、难记忆而实际上越来越用不到的语法。

(3) 本教材加强了数据库应用软件、数据库设计等方面的内容, 目的是让学生对数据库技术在整个应用软件开发过程中所处的环节、地位以及数据库应用程序的整体框架, 有一个全面的了解, 确立数据库技术的作用, 以便明确学习目的和意义。

(4) 本教材在最后两章安排了数据库应用程序的开发, 分别采用著名的数据库前台开发工具 PowerBuilder 语言和 C#语言, 通过实际应用系统的开发, 使学生们对数据库技术和数据库应用程序有全面、深刻的理解和掌握。第十章中完整、详细介绍的“学生成绩管理系统”软件开发, 具有广泛的迁移价值, 学生们只要更换数据库表数据, 就可以轻易地构成其他数据库应用系统。

由于时间仓促以及编写水平有限, 书中难免有错误和疏漏之处, 欢迎广大读者和同仁提出宝贵意见和建议。

编 者

参考授课计划或学习计划

章 名	项 目	任 务	安排课时
第一章 数据库基础	项目一 需求分析与数据库设计	任务一 针对“学生成绩管理系统”进行需求分析和数据库的概念设计	4
		任务二 针对“学生成绩管理系统”进行数据库的逻辑设计	4
	项目二 数据库应用程序与数据库管理系统的安装	任务 学会 SQL Server 2005 的安装、配置与简单使用	4
第二章 数据库和数据表	项目三 创建数据库和表	任务一 创建数据库	4
		任务二 数据表的物理设计	4
		任务三 表的创建、修改和删除	4
		任务四 表记录的创建、修改和删除	4
第三章 数据表查询	项目四 查询表中的数据	任务一 对数据表的简单查询	8
		任务二 对数据表的复杂查询	8
		任务三 表数据的排序与汇总	8
第四章 视图和游标	项目五 创建并使用视图	任务 视图的创建和查询	4
	项目六 声明并使用游标	任务 游标的使用	4
第五章 T-SQL 程序设计	项目七 在数据库系统中编程	任务一 常量、变量、自定义数据类型及运算符的使用	4
		任务二 流程控制语句的使用	4
		任务三 系统函数的使用	4
以上为第一个学期的教学参考课时量，共 72 课时			
第六章 索引与数据完整性	项目八 索引的创建和删除	任务 索引的创建和删除	4
	项目九 数据约束和数据完整性	任务一 设置默认值约束、unique 约束、check 约束和 identity 列属性	4
		任务二 创建主键、外键并实现参照完整性	4
第七章 存储过程和触发器	项目十 存储过程的创建与使用	任务 存储过程的创建与使用	4
	项目十一 触发器的创建与使用	任务 触发器的创建与使用	4
第八章 数据库的数据管理	项目十二 数据的导入与导出	任务 数据的导入与导出	4
	项目十三 数据库的备份与还原	任务 数据库的备份与还原	8

章 名	项 目	任 务	安排课时
第九章 数据库的安全管理	项目十四 两种身份验证模式下建立系统账户	任务一 Windows 身份验证模式创建 SQL Server 系统的账户	2
		任务二 混合身份验证模式创建 SQL Server 系统的账户	2
	项目十五 建立数据库用户和角色	任务 为数据库 STUDY 创建数据库用户、数据库角色等	4
	项目十六 给角色和用户授权	任务 给角色和用户授予权限	4
第十章 使用 PB/SQL Server 开发“学生成绩管理系统”	项目十七 利用开发工具 PB 构建一个完整的数据库系统	任务一 实现 PB 开发环境与数据库的静态连接	4
		任务二 实现应用程序与数据库的动态连接	4
	项目十八 设计开发“学生成绩管理系统”——添加记录模块	任务一 窗口设计、控件使用完成系统总体框架	4
		任务二 开发程序实现记录的滚动、添加和更新	4
	项目十九 设计开发“学生成绩管理系统”——删除记录模块	任务 事件中编程实现选中记录的删除	4
	项目二十 设计开发“学生成绩管理系统”——查询成绩模块	任务一 利用带参数数据对象实现多表复合查询功能	4
		任务二 排序功能和 SQL 语句的灵活运用	2
	项目二十一 设计开发“学生成绩管理系统”——成绩图形显示模块	任务 成绩图形显示的设计和实现	2
第十一章 使用 C#/SQL Server 开发“学生成绩管理系统”	项目二十二 利用 C#语言操作数据库	任务一 建立与数据库的连接	4
		任务二 复杂绑定实现数据查询	4
		任务三 简单绑定实现数据查询	4
		任务四 条件查询与数据编辑	4
	项目二十三 “学生成绩管理系统”的开发	项目实施	12
以上为第二个学期的教学参考课时量, 共 72 课时 (第十章和第十一章可以任选)			
本课程需两个学期完成, 总计参考课时量为 144 课时			

目 录

第一章 数据库基础	1
项目一 需求分析与数据库设计	1
任务一 针对“学生成绩管理系统”进行需求分析和数据库的概念设计	8
任务二 针对“学生成绩管理系统”进行数据库的逻辑设计	10
项目二 数据库应用程序与数据库管理系统的安装	11
任务 学会 SQL Server 2005 的安装、配置与简单使用	18
小结与习题	25
第二章 数据库和数据表	27
项目三 创建数据库和表	27
任务一 创建数据库	29
任务二 数据表的物理设计	32
任务三 表的创建、修改和删除	35
任务四 表记录的创建、修改和删除	39
小结与习题	42
第三章 数据表查询	43
项目四 查询表中的数据	43
任务一 对数据表的简单查询	47
任务二 对数据表的复杂查询	53
任务三 表数据的排序与汇总	60
小结与习题	64
第四章 视图和游标	66
项目五 创建并使用视图	66
任务 视图的创建和查询	67
项目六 声明并使用游标	72
任务 游标的使用	73
小结与习题	76
第五章 T-SQL 程序设计	78
项目七 在数据库系统中编程	78
任务一 常量、变量、自定义数据类型及运算符的使用	79
任务二 流程控制语句的使用	85
任务三 系统函数的使用	88
小结与习题	92

第六章 索引与数据完整性	93
项目八 索引的创建和删除	93
任务 索引的创建和删除	95
项目九 数据约束和数据完整性	101
任务一 设置默认值约束、unique 约束、check 约束和 identity 列属性	103
任务二 创建主键、外键并实现参照完整性	107
小结与习题	110
第七章 存储过程和触发器	111
项目十 存储过程的创建与使用	111
任务 存储过程的创建与使用	112
项目十一 触发器的创建与使用	118
任务 触发器的创建与使用	119
小结与习题	123
第八章 数据库的数据管理	125
项目十二 数据的导入与导出	125
任务 数据的导入与导出	126
项目十三 数据库的备份与还原	134
任务 数据库的备份与还原	136
小结与习题	145
第九章 数据库的安全管理	147
项目十四 两种身份验证模式下建立系统账户	147
任务一 Windows 身份验证模式创建 SQL Server 系统的账户	149
任务二 混合身份验证模式创建 SQL Server 系统的账户	151
项目十五 建立数据库用户和角色	153
任务 为数据库 STUDY 创建数据库用户、数据库角色等	156
项目十六 给角色和用户授权	161
任务 给角色和用户授予权限	162
小结与习题	166
第十章 使用 PB/SQL Server 开发“学生成绩管理系统”	168
项目十七 利用开发工具 PB 构建一个完整的数据库系统	168
任务一 实现 PB 开发环境与数据库的静态连接	169
任务二 实现应用程序与数据库的动态连接	174
项目十八 设计开发“学生成绩管理系统”——添加记录模块	181
任务一 窗口设计、控件使用完成系统总体框架	182
任务二 开发程序实现记录的滚动、添加和更新	185
项目十九 设计开发“学生成绩管理系统”——删除记录模块	190
任务 事件中编程实现选中记录的删除	191
项目二十 设计开发“学生成绩管理系统”——查询成绩模块	196
任务一 利用带参数数据对象实现多表复合查询功能	200

任务二 排序功能和 SQL 语句的灵活运用.....	203
项目二十一 设计开发“学生成绩管理系统”——成绩图形显示模块.....	205
任务 成绩图形显示的设计和实现.....	207
小结与习题.....	208
第十一章 使用 C#/SQL Server 开发“学生成绩管理系统”	210
项目二十二 利用 C#语言操作数据库	210
任务一 建立与数据库的连接	212
任务二 复杂绑定实现数据查询.....	215
任务三 简单绑定实现数据查询.....	219
任务四 条件查询与数据编辑	222
项目二十三 “学生成绩管理系统” 的开发	228
小结与习题.....	249
附录	251
参考文献	253

第一章 数据库基础

项目一 需求分析与数据库设计



【项目需求】

数据库是应用软件存放数据的地方，若要开发数据库应用程序，必须首先设计好数据库。

某学校有众多学生，划分若干专业，开设有许多课程，每一学生可选修多门课程。现需开发应用程序“学生成绩管理系统”，请进行需求分析，并完成数据库的设计。



【完成项目的条件】

- (1) 学会数据库需求分析的方法，能够准确找出实体及其属性。
- (2) 掌握数据库的概念设计方法，能够画实体联系 E-R 图。
- (3) 掌握数据库设计规范的三个范式。
- (4) 掌握数据库的逻辑设计方法，正确处理实体之间的关系。



【方案设计】

要完成“学生成绩管理系统”的数据库设计，首先需要对该系统进行详细的需求分析，针对客户提出的需求和需求调查的结果，将所涉及的各种信息进行归纳和分类，以确定这些信息名词中哪些可以作为实体，哪些作为实体的属性；其次，进行概念设计，将实体和属性用图形画出来，并加入必要的元素构成 E-R 图，继而确定实体之间的关系，再利用数据库设计的三个范式来检验概念设计是否合理。最后，运用数据库逻辑设计的方法将概念设计的结果转化为逻辑设计的结果，即关系表。

完整的数据库设计还包含数据库的物理设计，由于需要用到后面的知识，本项目先不涉及，留待后面补充。



【相关知识和技能】

一、数据库技术概述

随着社会生产技术的不断发展，人类所面对的信息量越来越大，如何快速高效地处理数据信息是摆在人们面前的一项重要课题。伴随着计算机技术快速发展起来的数据库技术已经成为数据处理的有力工具，成为信息技术中的一个重要支撑。

数据处理技术经历了人工管理——文件系统——数据库系统的历程。早期的数据处理是依靠手工或算盘来进行的。以后逐步代之以计算器、手摇计算机和电动计算机，数据存储形

式也从手写和纸质打孔, 逐渐发展到数据文件系统。20 世纪 60 年代后, 随着数据库技术的出现, 数据存储采用了规范高效的数据库形式。高效率存储设备的广泛使用, 使数据处理工作发生了革命性的改变, 计算机应用也从科学研究部门逐渐渗透到了各行各业。

数据库技术主要用于各种数据处理, 即对各种形式的数据进行收集、组织、加工、存储、抽取和传播等工作。其主要目的是从大量的、杂乱无章的甚至是难以理解的数据中抽取, 并加工推导出具有特定价值的数据, 从而为进一步的活动提供决策依据。可以这样认为: 没有数据库技术, 人们在浩瀚的信息世界中将显得手足无措。

至 21 世纪的今天, 利用数据库技术开发的数据库应用程序(人们常称为应用软件)已经无处不在。哪里有大量的数据信息处理的需要, 哪里就会出现数据库技术的身影。现在的数据库技术已经发展成为一门以数据库管理系统为核心, 内容丰富、领域宽广的新学科。它广泛地渗透到了社会方方面面的应用领域当中, 正发挥着巨大的数据处理和组织决策作用。数据库应用程序的开发, 更带动了一个巨大的软件产业的发展。

掌握了数据库技术以及开发方法, 我们不仅可以利用数据库管理系统方便地操作数据, 实现数据和使用者的有效管理, 胜任作为一名合格的数据库管理员的职责, 而且, 还可以在广泛的领域就业, 从事数据的信息化开发和管理的工作, 如为企业开发、维护数据库应用程序; 帮助企业挖掘信息, 实现信息的有效利用并辅助决策; 解决企业的信息孤岛问题, 实现数据接口间相互转换的难题, 等等。

二、数据库与数据库管理系统

数据库技术的基础是数据库(DataBase, DB)和数据库管理系统(DataBase Management System, DBMS)。

数据库是存放数据的仓库, 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上。数据库中的数据不仅包含数字和文字, 还可以包含图像、时间、音频及视频等。

例如, 把一个学校的学生、课程及学生成绩等数据, 按照一定的关系, 一定的格式组织在一起, 存放在计算机内, 就可以构成一个数据库。

因此, 数据库(DB)是长期存储在计算机内有结构的共享数据的集合。这些数据存储在计算机的存储介质中, 不会随计算机关机或死机而丢失, 它可以由各种用户共享, 并具有较强的冗余度和较高的数据独立性。

数据库管理系统(DBMS)是位于用户与操作系统之间的一个以统一的方式管理和维护数据库(DB)中数据的一系列软件的集合。

数据库与数据库管理系统的关系如图 1-1 所示。

数据库管理系统提供以下功能:

- 数据定义功能: 可定义数据库中的数据对象。
- 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。
- 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。
- 数据库的安全保护功能: 保证只有赋予权限的用户才能访问数据库中的数据。
- 数据的并发控制功能: 使多个应用程序可在同一

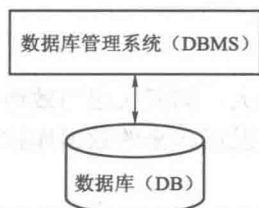


图 1-1 数据库与数据库管理系统的关系

时刻同时访问数据库中的数据。

- 数据库系统的故障恢复功能：使数据库运行出现故障时进行数据库恢复，以保证数据库可靠运行。

- 在网络环境下访问数据库的功能。

- 方便有效地存取数据库信息的接口和工具。编程人员使用软件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。

自 20 世纪 70 年代提出数据的关系模型后，商用数据库系统迅速采用了这种模型，涌现出了很多优良的关系数据库管理系统（Relational DataBase Management System, RDBMS），并成为市场上的主导产品，主流的大型关系型数据库管理系统包括 Oracle、Sybase、SQL Server、DB2、Informix、Ingres 等，小型的关系型数据库管理系统包括 MySQL、Access、Visual FoxPro、Adaptive server anywhere 等。

三、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

层次模型是指以树形层次结构组织数据，它属于格式化数据模型。图 1-2 所示为某学校层次模型组织的数据。

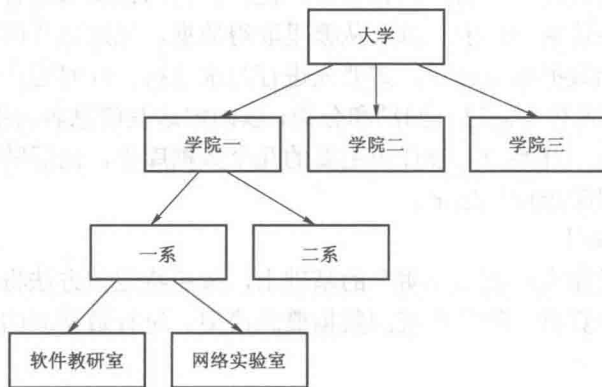


图 1-2 按层次模型组织的数据示例

2. 网状模型

网状模型又叫网络模型，它也属于格式化数据模型。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。这样的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，每个部门都销售所有 5 种产品。

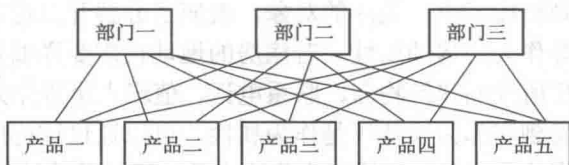


图 1-3 按网状模型组织的数据示例

3. 关系模型

关系模型是指以二维表格（关系表）的形式组织数据库中的数据，它不同于格式化模型的风格和理论基础，是一种数学化的模型。关系模型的基本组成是关系。按关系模型组织的数据示例见表 1-1。

表 1-1 按关系模型组织的数据示例

学号	姓名	性别	出生时间	专业	总学分	备注
081101	王林	男	1990-2-10	计算机	50	
081103	王燕	女	1989-10-6	计算机	50	
081108	林一帆	男	1989-8-5	通信工程	52	已提前修完一门

这张二维表显示了每个学生的情况。表的每一行是一个记录，也称为一个元组。每一列是记录中的一个字段，表示其中的一个属性。

四、数据库设计

数据库的设计建立在详细的需求分析基础上，随后还必须经过三个过程，即概念设计、逻辑设计和物理设计。

1. 需求分析

需求分析是指根据对客户的需求调查和客户提供的需求说明书进行判断和分析，确定客户需要利用何种信息达成何种功能，以及从哪里取得数据，经过怎样的处理，最后以怎样的形式反映出来等。开始数据库设计前，需要先进行需求分析。针对客户提出的需求和需求调查的结果，将所涉及的各种信息进行归纳和分类，以确定这些信息名词中哪些可以作为实体，哪些作为实体的属性。一开始应先设计出主要的几个数据库表，而后随着需求分析的深入，再将一些辅助性的数据库表设计出来。

2. 数据库的概念设计

概念设计的任务是在客户需求说明书的基础上，按照特定的方法将客户的需求抽象为一个不依赖于任何具体计算机和软件系统的数据概念模型。进行数据库的概念设计需要了解以下知识。

1) 实体、属性和实体集

实体是指独立存在并且可以相互区别的事物。属性是从属于实体的、用于描述实体特性的信息。实体的具体个体集合称为实体集。

在现实中有时是很难区分哪个是实体哪个是属性的。一个概念名词可能是实体也可能只作为属性，这取决于它是否是重点关注的对象。比如，医院的病房，对于病人这个实体来说，它是病人的属性（尽管病房这个属性还能再分，不符合下文中提到的第一范式）。但如果要对病房进行专门的管理，则变成了重点关注的对象，就应当分解开来把它作为实体，而将病房专业、地址、联系电话等作为病房的属性。若病房的地址也需要详细管理起来，则病房地址也应变成实体，再将病房所处的建筑物名、联系电话、值班人员等作为属性。

判断一个事物或信息到底作为实体还是作为属性，可以通过这样的原则：凡是需求分析中需要重点关心的独立事物，就可以直接把它作为实体，而某种单纯的特性，如时间、地点、

数量、大小、长度等，则通常只作为属性。

2) 实体间的三种关系

一对一关系，是指一个具体的实体只对应另一个具体的实体。如：某丈夫只对应某妻子；某班级只对应一个班长。

一对多关系，是指一个具体的实体对应（或属于）多个其他具体的实体。如：某工厂包含多个车间；某班级有很多学生。

多对多关系，是指多个具体的实体对应多个其他具体的实体。如：学生与选修课程的关系；多个连锁店与众多销售商品的关系。在本例子中，我们确认学生实体与课程实体的关系是多对多的关系，因为一个学生可以选修多门课程，而一门课程也可以被多个学生选修。

3) E-R 图和主键

E-R 图即实体 (Entity)-联系 (Relational) 图，其画法是：将实体用矩形框画出来，属性用圆角矩形表示，实体和其属性用线段连接起来，实体间关系则用菱形来连接。

在实体的所有对应属性中，如果某属性的值或最小属性组合的值能唯一标识（即唯一区分）该实体，则将该属性或属性组合称为键，对于每一个实体集，可指定其中一个键为主键。

4) 规范数据库概念设计的三个范式

第一范式：每个实体的元组中每一个属性都不可再分。

如：

学校建筑
• 建筑号
• 名称
• 地址
@ 教室

由于教室这个属性可以再分，不符合第一范式的规范，因此应分解为：

学校建筑
• 建筑号
• 名称
• 地址

+

教室
• 教室编号
• 楼层
• 面积

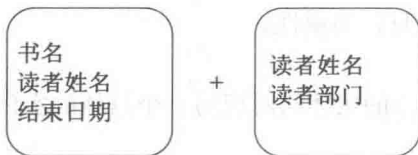
教室变成单独的实体，两元组都符合。

第二范式：每个实体的元组中不能存在与主关键字无关的属性。

如：

书名
读者姓名
读者部门
结束日期

由于读者部门与书名无关，不符合第二范式的规范，因此应分解为：

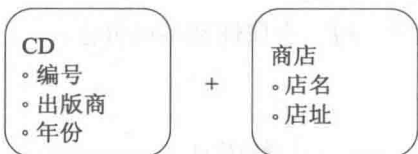


第三范式：每个实体的元组中非关键字属性间不存在依赖关系。

如：



由于店名和店址均是 CD 的提供来源，店名和店址之间存在依赖关系，不符合第三范式的规范，因此应分解为：



5) 数据库概念设计的方法

数据库概念设计的方法可以归纳为：

- 从需求分析中提取出所有的实体和属性。
- 分析每个实体，确定实体间是否存在关系。有关系的话，判断是属于一对一关系，还是一对多关系，还是多对多关系。
- 将实体、属性及实体之间的关系按照规范画出 E-R 图。
- 在 E-R 图中为实体添加主键标识。
- 按照三个范式对概念设计进行校验修正。

3. 数据库的逻辑设计

数据库的逻辑设计，是以 E-R 图为基础，根据实体与属性的联系或实体与实体之间的关系，实现概念模型向逻辑模型的转换。具体地讲，是要确定数据库中应包含哪些表，每个表又有怎样的结构。

逻辑设计的方法，首先就是把每一个实体及其属性设计为一个表，定义表名，实体的属性作为表中的字段，找到主键并用下划线表示出来，其形式为：

表名 (属性 1, 属性 2, 属性 3, ...)

其次，处理实体之间的关系，具体需要遵循以下原则：

- (1) 若表 A 和表 B 是一对一关系：可以合并，把表 A 的主键加到表 B 中当外键。
- (2) 若表 A 和表 B 是一对多关系：必须把表 A 的主键加到表 B 中当外键。
- (3) 若表 A 和表 B 是多对多关系：除了生成表 A 和表 B 外，还要生成一个关系表：表 A 主键+表 B 主键+多对多关系中自己的属性。

注：外键指表 B 中含有表 A 中的主键列，则在表 B 中称为外键。

实际的数据库设计工作中，还会涉及数据库冗余的问题和数据库查询时的复杂性和效率问题。数据库冗余是指构成数据库的表中存在大量被浪费的存储空间。例如，以下的逻辑模型就存在大量数据库冗余，见表 1-2 中的阴影部分。

表 1-2 系别字段产生冗余

学号	姓名	性别	出生时间	专业	系别	总学分	备注
081101	王林	男	1990-2-10	计算机	信息系	50	提前修完一门
081102	袁芳	女	1990-8-24	计算机	信息系	52	
081301	张金玲	女	1991-2-16	信息管理	信息系	56	提前修完一门
081304	李晓军	男	1991-1-4	通信工程	信息系	54	

上表 1-2 中，系别字段产生大量冗余，按照第三范式的规范，专业与系别存在依赖关系，应该拆分成两个表，就可以消除冗余，即将数据库表 1-2：

tb_student1 (学号, 姓名, 性别, 出生时间, 专业, 系别, 总学分, 备注)

拆分成以下两个表：

tb_student (学号, 姓名, 性别, 出生时间, 专业, 总学分, 备注)

tb_specialty (专业名称, 系别)

这样，冗余就消除了。

是否表 1-2 的设计不可取呢？实际情况并非如此。有时具有冗余的设计反而会带来软件开发的便利以及较小的软件维护成本。在表 1-2 中当需要查询某个系有多少学生时，只需一条简单的查询语句便可解决。而完全按照三个范式进行的数据库设计，即消除了冗余后，则需要两个表的组合查询才能达到目的。这里的冗余虽然浪费了存储空间，但减少了开发的复杂性和维护成本。因此，数据库的冗余问题需具体情况具体分析。

优秀的数据库设计并非是最恰当的设计。纵观涉及计算机的所有设计，均离不开存储空间和运算效率这两方面的权衡。当为用户进行数据库设计和软件开发时，不可避免地涉及用户一方存储设备的投资预算的限制以及对软件运行效率的要求。对于软件公司一方来说，节省开发和未来维护的成本是其首要考虑的因素。这些因素都影响着数据库的最终设计方案。

综上所述，并非冗余最小的设计是最好的。数据库设计的三个范式也并非必须遵守的金科玉律。怎样的数据库设计才是最佳的，需要根据实际情况进行综合评判。

4. 数据库的物理设计

数据库最终是要存储在物理设备上的。为一个给定的逻辑数据模型选取一个最适合应用环境的物理结构（存储结构与存取方法）的过程，就是数据库的物理设计。数据库的物理结构依赖于给定的 DBMS 和硬件系统。因此设计人员必须充分了解所用 DBMS 的内部特征，特别是存储结构和存取方法；充分了解应用环境，特别是应用的处理频率和响应时间要求；以及充分了解外存设备的特性。

具体来讲，数据库物理设计需要确定数据库存储结构、设计数据的存取路径，确定数据的存放位置，确定系统配置，最后需要对时间效率、空间效率、维护代价和各种用户要求进行权衡。涉及表字段的数据类型、表的索引方法、数据库日志和备份的存储安排及数据库系

统配置参数等。后面章节将会逐步深入学习。

任务一 针对“学生成绩管理系统”进行需求分析和数据库的概念设计

【任务目标】

- (1) 对“学生成绩管理系统”进行需求分析，确定实体及其属性。
- (2) 掌握数据库的概念设计方法，画出 E-R 图。
- (3) 掌握数据库设计规范的三个范式。

【任务分析】

要设计数据库必须先对“学生成绩管理系统”进行需求分析。了解这个系统需要解决什么问题，达到什么功能。借此了解需重点关注的信息是哪些。

本系统需要对学生进行管理，那么学生是重点关注的信息，可以作为实体。本系统还需要对课程进行管理，那么课程就是实体。这样一下子就抓住了问题的主要矛盾，再以这两个实体为中心，就可以很快找出各自的属性。

浏览各实体对应的属性，观察这些属性是否可以涵盖本系统的用户需求，确认这些属性及其综合运算可以达成用户所需的功能。如若不够，则需再增加属性甚至增加实体。

将所有实体和属性用 E-R 图画出来。并考察各实体间的关系，在 E-R 图上标示出来。再分别寻找每个实体集的主键，用斜线标注在 E-R 图上。

用三个范式对 E-R 图中的概念设计结果进行检查，权衡冗余，进行综合判断，确保数据库的概念设计最适合实际情况。

【知识准备】

记录——表的每一行称为一条记录。

元组——表的每一行也称为一个元组。

字段——每一列是记录中的一个字段。

实体——是指独立存在并且可以相互区别的事物。

属性——是从属于实体的用于描述实体特性的信息。

实体集——某实体的具体集合就是实体集。

E-R 图——用线段将实体与其从属的特性联系起来所形成的关系图，称作 E-R 图。

主键——每一个实体集中，能唯一区分实体的某属性或最小属性组合，称为主键。

数据库概念设计的方法：

- (1) 从需求分析中提取出所有的实体和属性。
- (2) 分析每个实体，确定实体间是否存在关系。有关系的话，判断是属于一对一关系，一对多关系，还是多对多关系。

(3) 将实体、属性及实体之间的关系按照规范画出 E-R 图。

(4) 在 E-R 图中为实体添加主键标注。

(5) 对照三个范式，看概念设计是否符合范式规范，否则重新设计。

规范数据库概念设计的三个范式：