

 高等教育规划教材

SQL Server 2014 数据库教程

秦婧 傅冬 王斌 编著



免费提供电子教案

下载网址 <http://www.cmpedu.com>



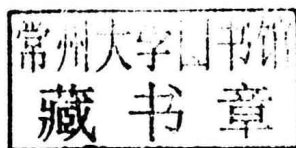
机械工业出版社
CHINA MACHINE PRESS



高等教育规划教材

SQL Server 2014 数据库教程

秦 婧 傅 冬 王 斌 编著



机械工业出版社

本书主要介绍了数据库基础知识、SQL Server 2014 的安装及企业管理器的使用、SQL 语句的分类、数据库中常用的对象创建和维护(包括数据库、表、视图、函数以及存储过程、触发器)、数据库的安全性、数据库的备份与还原等,并在最后一章配以实例讲解如何使用 C#语言连接 SQL Server 2014 数据库。本书在每章的前面列出了学习目标,以方便读者对本章涉及的内容有所了解;在每章的后面都有本章习题,读者可以将其作为考核本章知识点的复习题。本书在讲解理论的同时,注重将理论联系实践,以实例的方式演练每一个知识点。

本书既可以作为高等院校计算机软件技术课程的教材,也可以作为管理信息系统开发人员的技术参考书。

本书配有授课电子课件,需要的教师可登录 www.cmpedu.com 免费注册,审核通过后下载,或联系编辑索取(QQ: 286726866, 电话: 010 - 88379742)。

图书在版编目(CIP)数据

SQL Server 2014 数据库教程/秦婧编著. —北京:机械工业出版社,2017.6
高等教育规划教材

ISBN 978-7-111-57384-5

I. ①S… II. ①秦… III. ①关系数据库系统-高等学校-教材
IV. ①TP311.138

中国版本图书馆 CIP 数据核字(2017)第 175068 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

策划编辑:郝建伟 责任编辑:郝建伟 范成欣

责任校对:张艳霞 责任印制:李 昂

北京宝昌彩色印刷有限公司印刷

2017 年 8 月第 1 版·第 1 次印刷

184 mm × 260 mm · 20.75 印张 · 505 千字

0001-3000 册

标准书号:ISBN 978-7-111-57384-5

定价:55.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

电话服务

网络服务

服务咨询热线:(010)88379833

机工官网:www.cmpbook.com

读者购书热线:(010)88379649

机工官博:weibo.com/cmp1952

教育服务网:www.cmpedu.com

封面无防伪标均为盗版

金书网:www.golden-book.com

出版说明

当前,我国正处在加快转变经济发展方式、推动产业转型升级的关键时期。为经济转型升级提供高层次人才是高等院校最重要的历史使命和战略任务之一。高等教育要培养基础性、学术型人才,但更重要的是加大力度培养多规格、多样化的应用型、复合型人才。

为顺应高等教育迅猛发展的趋势,配合高等院校的教学改革,满足高质量高校教材的迫切需求,机械工业出版社邀请了全国多所高等院校的专家、一线教师及教务部门,通过充分的调研和讨论,针对相关课程的特点,总结教学中的实践经验,组织出版了这套“高等教育规划教材”。

本套教材具有以下特点:

1) 符合高等院校各专业人才的培养目标及课程体系的设置,注重培养学生的应用能力,加大案例篇幅或实训内容,强调知识、能力与素质的综合训练。

2) 针对多数学生的学习特点,采用通俗易懂的方法讲解知识,逻辑性强、层次分明、叙述准确而精炼、图文并茂,使学生可以快速掌握、学以致用。

3) 凝结一线骨干教师的课程改革和教学研究成果,融合先进的教学理念,在教学内容和方法上做出创新。

4) 为了体现建设“立体化”精品教材的宗旨,本套教材为主干课程配备了电子教案、学习与上机指导、习题解答、源代码或源程序、教学大纲、课程设计和毕业设计指导等资源。

5) 注重教材的实用性、通用性,适合各类高等院校、高等职业学校及相关院校的教学,也可作为各类培训班教材和自学用书。

欢迎教育界的专家和老师提出宝贵的意见和建议。衷心感谢广大教育工作者和读者的支持与帮助!

机械工业出版社

前 言

SQL Server 2014 数据库是目前受众比较多的一款数据库产品。之所以受到广大用户的青睐，一方面是它的易学易用，让初学者能够很快上手；另一方面是由于它与 Visual Studio 平台有很好的整合性，能够方便用户在 Visual Studio 平台上使用 C#或 C++ 语言开发数据库软件。此外，随着数据分析的兴起，SQL Server 2014 数据库也提供了相应的数据挖掘功能，方便用户快速分析数据。

本书针对以往 SQL Server 教材中存在的一些问题（如专业性太强、版本相对滞后、理论与实践脱节等），以初级读者为对象，按照知识的体系结构和读者的特点，逐步深化知识点。本书可以引导读者快速掌握 SQL Server 的基本语法以及数据表的操作、存储过程和触发器的使用，进而完成用 C#语言连接 SQL Server 数据库的操作。本书内容设置由浅入深，同时结合实际操作步骤以及完整的案例项目，并附有示例代码，注重理论与实践相结合。

在内容编写上，本书以 SQL Server 2014 为版本讲解 SQL 语句，首先介绍了 SQL Server 中的基本语法、操作表、视图及函数等基础内容，然后介绍了存储过程、触发器及权限管理，最后介绍了使用 C#语言连接 SQL Server 数据库，涵盖了 SQL Server 从初学到进阶的所有主要内容。

全书共分为 12 章，各章具体内容如下。

- 第 1 章：概括地介绍了数据库基础知识、SQL Server 2014 的安装及 SQL Server 的服务管理。
- 第 2 章：主要讲解了 SQL 语句的分类，以及数据库的创建和维护操作。
- 第 3 章：主要介绍了表的基本概念，以及创建和维护表的操作。
- 第 4 章：主要讲解了如何操作表中的数据，包括向表中添加数据、修改数据及删除数据。此外，还介绍了如何给表设置约束及管理约束。
- 第 5 章：主要讲解了查询操作，包括多表查询、分组查询和子查询等操作。
- 第 6 章：主要讲解了函数的操作，包括系统函数和自定义函数的创建及维护。
- 第 7 章：主要讲解了视图的操作，包括视图的基本概念、创建视图及维护视图。
- 第 8 章：主要讲解了 T-SQL 语句的使用，包括变量的声明、条件语句和循环的使用，并介绍了游标的操作。
- 第 9 章：主要讲解了存储过程和触发器的使用，包括存储过程的创建及维护、触发器的创建及维护等。
- 第 10 章：主要讲解了数据库中用户、角色和权限的操作，包括用户的创建和维护、角色的创建和应用以及权限的授予和收回等。
- 第 11 章：主要讲解了数据库的备份和还原，包括使用命令和企业管理器对数据库进

行备份、还原、分离和附加的操作。

- 第 12 章：主要讲解了使用 C# 语言连接 SQL Server 2014 的操作，包括 ADO.NET 的使用，以及数据库连接类的编写、用户登录注册模块、音乐管理模块的实现等。

本书是由秦婧、傅冬和王斌共同编写的。在本书的编写过程中，得到了同行的支持和帮助，在此一并表示感谢。

由于编者水平有限，书中难免存在不妥之处，敬请广大读者原谅，并提出宝贵意见。编者邮箱：56981673@qq.com。

编 者

目 录

出版说明

前言

第1章 走进 SQL Server 1

1.1 数据库基础知识 1

1.1.1 了解相关概念 1

1.1.2 认识数据模型 2

1.1.3 数据库的设计 4

1.2 了解 SQL Server 7

1.2.1 SQL Server 的发展历程 7

1.2.2 SQL Server 2014 的新特性 8

1.2.3 其他数据库产品 9

1.3 SQL Server 的安装 10

1.3.1 SQL Server 各版本简介 10

1.3.2 SQL Server 2014 的安装环境 13

1.3.3 SQL Server 2014 组件 14

1.3.4 SQL Server 2014 的安装 14

1.3.5 卸载 SQL Server 2014 20

1.3.6 安装实例数据库 23

1.4 图形界面管理器简介 25

1.4.1 连接 SQL Server 服务器 25

1.4.2 管理器菜单简介 26

1.5 SQL Server 的服务管理 27

1.6 本章小结 29

1.7 本章习题 29

第2章 管理数据库 31

2.1 SQL 简介 31

2.1.1 SQL 的分类 31

2.1.2 书写约定 32

2.2 创建数据库 33

2.2.1 基本语法 33

2.2.2 在指定位置创建数据库 34

2.2.3 使用数据库中的文件组 38

2.2.4 查看数据库 39

2.3 修改数据库 42

2.3.1 基本语法 42

2.3.2 管理数据库中的文件 43

2.3.3 管理文件组 45

2.3.4 管理数据库 45

2.4 删除数据库 46

2.5 使用企业管理器操作

数据库 46

2.5.1 创建数据库 46

2.5.2 管理数据库 48

2.6 综合实例：创建并操作音乐

播放器数据库 51

2.7 本章小结 52

2.8 本章习题 53

第3章 管理表 54

3.1 基本概念 54

3.2 创建表 55

3.2.1 数据类型 55

3.2.2 用户定义数据类型 58

3.2.3 创建永久表 60

3.2.4 创建临时表 61

3.3 维护表 62

3.3.1 查看表的信息 62

3.3.2 修改表 64

3.3.3 重命名表和删除表 66

3.4 使用企业管理器管理表 67

3.4.1 使用企业管理器创建表 68

3.4.2 使用企业管理器维护表 70

3.5 综合实例：向音乐播放器

数据库中添加表 71

3.6 本章小结 74

3.7 本章习题	74	5.3.2 使用分组查询	129
第4章 操作表中的数据	75	5.4 子查询	130
4.1 表中的约束	75	5.4.1 子查询中所用的运算符	131
4.1.1 约束的作用	75	5.4.2 单行子查询	131
4.1.2 主键约束	77	5.4.3 多行子查询	133
4.1.3 外键约束	80	5.5 综合实例：根据条件完成	
4.1.4 唯一约束	83	音乐播放器数据库	
4.1.5 检查约束	84	中的查询语句	134
4.1.6 默认值约束	86	5.6 本章小结	135
4.1.7 使用企业管理器管理约束	87	5.7 本章习题	135
4.2 运算符和表达式	92	第6章 函数	136
4.3 向表中添加数据	94	6.1 系统函数	136
4.3.1 添加数据的基本语法	94	6.1.1 数学函数	136
4.3.2 复制数据	96	6.1.2 字符串函数	142
4.4 维护表数据	97	6.1.3 日期函数	147
4.4.1 修改表中的数据	97	6.1.4 其他函数	150
4.4.2 删除表中的数据	98	6.2 自定义函数	155
4.5 使用企业管理器管理表中的		6.2.1 基本语法	155
数据	99	6.2.2 创建标量值函数	157
4.6 综合实例1：为音乐播放器		6.2.3 创建表值函数	159
中的数据表设置约束	100	6.3 维护自定义函数	160
4.7 综合实例2：管理音乐播放器		6.3.1 查看函数信息	160
中表的数据	101	6.3.2 重命名函数	161
4.8 本章小结	103	6.3.3 修改函数	162
4.9 本章习题	103	6.3.4 删除函数	163
第5章 查询	105	6.4 使用企业管理器管理自定义	
5.1 基本查询语句	105	函数	163
5.1.1 不带条件的查询	106	6.5 本章小结	165
5.1.2 带条件的查询	108	6.6 本章习题	166
5.1.3 模糊查询	111	第7章 视图	167
5.1.4 NULL 值查询	114	7.1 了解视图	167
5.1.5 查询结果排序	115	7.2 创建视图	168
5.2 多表查询	117	7.2.1 创建视图语法	169
5.2.1 笛卡儿积	118	7.2.2 创建视图的方法	170
5.2.2 内连接查询	118	7.2.3 创建加密视图	172
5.2.3 外连接查询	122	7.2.4 创建绑定视图	173
5.2.4 结果集的运算	124	7.2.5 使用企业管理器创建视图	174
5.3 分组查询	127	7.3 管理视图	177
5.3.1 聚合函数	127	7.3.1 查看视图信息	177

7.3.2	重命名视图	178	9.1.4	使用企业管理器操作存储 过程	225
7.3.3	修改视图	179	9.2	触发器	231
7.3.4	删除视图	179	9.2.1	触发器的作用及分类	231
7.3.5	使用企业管理器工具管理 视图	180	9.2.2	创建 DML 触发器	231
7.4	可更新视图	181	9.2.3	管理 DML 触发器	236
7.4.1	可更新视图的条件	181	9.2.4	使用企业管理器操作 触发器	239
7.4.2	通过视图增加记录	182	9.3	本章小结	242
7.4.3	通过视图修改和删除记录	183	9.4	本章习题	242
7.5	本章小结	184	第 10 章	数据库的安全性	243
7.6	本章习题	184	10.1	用户	243
第 8 章	T-SQL 基础	185	10.1.1	SQL Server 的身份验证模式	243
8.1	T-SQL 中的书写约定	185	10.1.2	创建用户	245
8.2	常量和变量	185	10.1.3	管理用户	246
8.2.1	常量	186	10.1.4	登录用户管理	247
8.2.2	变量	187	10.1.5	使用企业管理器管理用户	249
8.3	常用语句	190	10.2	角色	254
8.3.1	条件语句	190	10.2.1	了解常用的角色	254
8.3.2	循环语句	194	10.2.2	创建角色	255
8.3.3	控制语句执行时间	197	10.2.3	修改角色	256
8.3.4	异常处理	198	10.2.4	删除角色	258
8.4	游标	201	10.2.5	使用企业管理器操作角色	259
8.4.1	游标的作用	201	10.3	权限	264
8.4.2	声明游标	201	10.3.1	授予权限	265
8.4.3	打开和读取游标	202	10.3.2	收回权限	266
8.4.4	关闭和释放游标	206	10.3.3	拒绝权限	267
8.4.5	实例演练: 使用游标完成表 数据的读取和操作	207	10.3.4	使用企业管理器操作权限	268
8.5	使用事务控制语句	208	10.4	本章小结	270
8.5.1	事务的作用	208	10.5	本章习题	270
8.5.2	启动和保存事务	209	第 11 章	数据库的备份与还原	271
8.5.3	提交和回滚事务	210	11.1	数据库备份	271
8.6	本章小结	211	11.1.1	备份的类型与作用	271
8.7	本章习题	211	11.1.2	备份数据库	272
第 9 章	存储过程与触发器	212	11.1.3	备份日志文件	274
9.1	存储过程	212	11.1.4	备份文件和文件组	274
9.1.1	存储过程的分类	212	11.2	还原数据库	276
9.1.2	创建存储过程	214	11.2.1	还原数据库文件	276
9.1.3	管理存储过程	221	11.2.2	还原文件和文件组	277

11.3 使用企业管理器备份和还原		2014	296
数据库	278		
11.3.1 使用企业管理器备份			
数据库	278		
11.3.2 使用企业管理器还原			
数据库	280		
11.4 数据库的分离和附加	284		
11.4.1 数据库的分离	284		
11.4.2 数据库的附加	285		
11.5 使用企业管理器分离和附加			
数据库	285		
11.6 数据的导入和导出	287		
11.7 本章小结	295		
11.8 本章习题	295		
第 12 章 使用 C#连接 SQL Server			
12.1 了解 ADO.NET	296		
12.1.1 ADO.NET 中涉及的类	296		
12.1.2 使用 SqlDataReader 类操作			
数据	298		
12.1.3 使用 DataSet 类操作数据	299		
12.2 音乐播放器的设计与实现	300		
12.2.1 数据库与连接类设计	300		
12.2.2 功能设计	305		
12.2.3 用户登录注册模块的实现	306		
12.2.4 普通用户操作模块的实现	308		
12.2.5 管理员用户操作模块的			
实现	313		
12.3 本章小结	320		
参考文献	321		

第1章 走进 SQL Server

SQL Server 是一款源于 20 世纪 80 年代的数据库软件，至今仍处于主流地位。它与 Windows 平台紧密结合，提供了可靠的企业级信息管理系统方案，是数据库开发者应该学习的数据库软件之一。本章学习目标如下。

- 了解数据库的概念。
- 了解数据库模型的分类。
- 初步掌握 E-R 图。
- 了解 SQL Server 的发展。
- 了解 SQL Server 2014 的新特性。
- 了解 SQL Server 2014 各版本的差异。
- 了解 SQL Server 的安装环境。
- 掌握 SQL Server 2014 的安装和卸载。
- 初步掌握 SQL Server Management Studio 工具。
- 初步掌握配置管理器。

1.1 数据库基础知识

随着科技发展和社会进步，各领域逐渐细化。同时，相关的信息收集量在不断扩大，信息的价值也在不断提升。传统的纸质记录和检索方式已经不能满足人们的需求，于是出现了以计算机文件保存数据的方式。为了使信息价值最大化，有些公司就以固定的结构来组织这些数据，以达到让用户从不同角度对数据进行查询的目的。这就是最初的数据库，它可以使人们更加便捷地存储、查找想要的数据库。

1.1.1 了解相关概念

在学习数据库的初期，读者会接触到以下两个概念：一个是数据库（Database），另一个是数据库管理系统（Database Management System, DBMS）。

数据库就是按照一定模型和结构进行组织并存储在介质中的数据集合。这些数据并不是杂乱的，而是有组织的，尽量剔除了无用的、错误的信息以及不必要的冗余。同时，数据允许共享，允许数据库管理员遵循统一的标准对其进行管理控制，包括增加、删除、修改和查询等；也允许应用程序对其进行相似效果的操作。

例如，以图书阅览室中书籍的属性以及借阅情况创建数据库，其中包括书籍编号、书籍分类、ISBN、书名、出版社、作者、数量、位置、是否借出等。管理者使用这个数据库可以针对各种属性进行查询，然后到指定的位置去拿书即可。除此之外，也可以对书籍做各方面的统计。例如，统计某段时间内，哪个出版社或谁的书更受欢迎等。这是传统图书馆做起

来很耗时的事情，然而利用数据库就可以快速完成，极大地提高了工作效率，降低了人工成本。

数据库管理系统就是用来管理数据库的大型软件，包含了创建、使用和维护数据库的方法和工具。SQL Server 本身就是一个数据库管理系统，它可以创建多个数据库实例。用户可以通过它提供的工具来访问、管理和维护数据库实例的数据。

 注意：数据库中的数据并不是单纯的数值，它可以是数值，也可以是文字、图片或音乐、视频等信息。

1.1.2 认识数据模型

在数据库管理系统中，经典的数据模型有 3 种，分别是层次模型、网状模型和关系模型。其中，层次模型和网状模型通常被早期的数据库使用，诞生于 20 世纪 60 年代，属于第一代数据库；关系型数据库诞生于 20 世纪 70 年代，属于第二代数据库。它们的特性和差异介绍如下。

1. 层次模型

层次模型是基于“上级/下级”模式的，整体表现为树形结构，如图 1-1 所示。图 1-1 中的每一个结点都表示一种记录类型（可以包含一个或多个记录值），不同的记录类型不可以重名；每个记录类型允许有多个字段，字段之间不允许重名。如果要访问下层（如层 2）的数据，则需要通过它上级（如层 1）中的指针记录。如果上级不存在该记录，则下层不存在与之对应的记录。对于层次模型，读者需要了解以下几点。

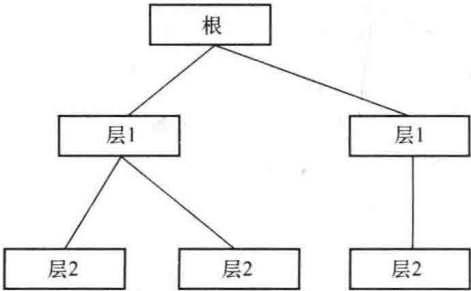


图 1-1 层次模型

- 层次数据模型中最基本的数据关系是层次关系。
- 有且只有一个结点没有父结点，该结点是根结点。
- 除了根结点外，其他结点有且只有一个父结点。
- 一个结点可有多个子结点，表示一对多的关系。

如果要存取某个记录类型中的数据，则需要从根结点开始，按照树的层次，依据记录指针逐层向下进行。例如，在图 1-2 中查找某个客户最近购买了哪些商品，需要先从“商品订单”开始，找到订单列表中的客户 ID 和其对应的商品 ID；然后根据客户 ID 在“客户信息”中找到客户名称；最后在“商品信息”中，根据商品 ID 获取商品信息。这样就获取了某个客户购买商品的信息。

层次模型适合处理一对多的关系。假如售货的机构有两种销售方式：一种是直销，图 1-2 的模式就是胜任的（不包含虚线分支）；另一种是代售（客户可由代理商购买商品），则图 1-2 就是无法胜任的（包含虚线分支）。因此，要想使层次模型完美地描述共存的销售方式，就需要开发者对图 1-2 进行较大改动。

2. 网状模型

网状模型和层次模型比较相似，同样基于“上级/下级”模式，但它允许一个结点有多

个父结点和子结点。即网状模型不仅能处理一对多的关系，也能处理多对多的关系。网状结构访问数据不是必须从根结点开始，而是可以从任何结点开始遍历整个结构。利用网状结构描述共存的销售模式，如图 1-3 所示。

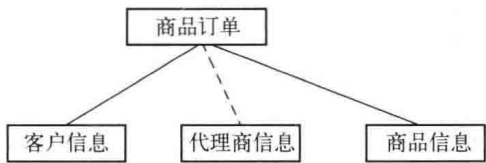


图 1-2 层次结构演示图



图 1-3 网状模型演示

虽然网状模型比层次模型有一定的优势，但是二者的灵活性都不是很好。当逻辑结构发生变化时，都可能导致重建数据库，这会降低开发效率，提高开发成本，不是开发者想要的。

3. 关系模型

关系模型比网状模型先进，由 IBM 公司的 E. F. Codd 博士在 1970 年提出，是基于谓词逻辑和集合论的一种模型。它的抽象级别相对较高，理解起来更简单、清晰，也方便使用。

关系模型的数据逻辑结构是一张二维表（见图 1-4 中的用户登录表），利用二维表来表示实体与实体间的联系。实际上，一个关系数据库会由一个或多个表组成。表由行和列组成（也可说记录和字段）。如果多个表中的某些列之间有相互依赖关系，则通过参照完整性来强制这种依赖性（见图 1-5），从而使表之间具有关联性。

与前两种模型相比，关系模型的优势如下：

- 1) 结构单一，容易理解。模型中实体以及它们之间的联系都用关系表示，关系对应了二维表，这种结构简单清晰。
- 2) 操作方便。用户只需使用查询语言就能对数据进行操作。
- 3) 安全、易于维护。完整性的使用（域完整性、实体完整性、参照完整性和用户自定义的完整性）降低了不必要的数据库冗余，保证了数据的有效性。

在关系模型中，读者需要了解以下几个概念。

- 关系：一个关系对应了一张二维表，可以认为表名就是关系名。
- 元组：可以看成二维表中的一行，也称为记录。
- 属性：可以看成二维表中的一列，也称为字段。
- 域：属性的取值范围。
- 候选码：关系中的一个属性组能唯一地标识一个元组，但其真子集不具有该能力，则把该属性组称为候选码。
- 主码：如果有多个候选码，在其中选择一个来做主码。这个选定的主码可以认为是主键。

以上概念同二维表各部分的对应关系如图 1-4 所示。图中是登录用户的表结构，名为“用户登录表”。

由于“用户登录表”中的“安全级别”字段中的值取自“用户安全级别表”中“安全编号”字段的值，而“安全编号”字段是它所在表的主键，因此这两张表是有主、外键关系的（参照完整性），如图 1-5 所示。

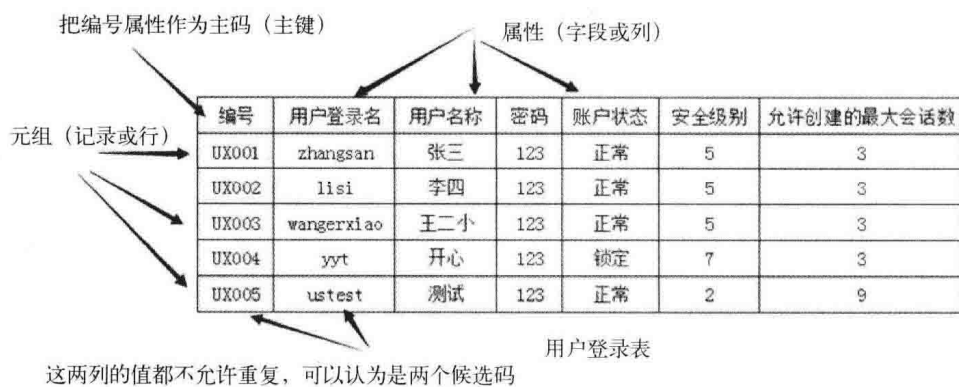


图 1-4 关系模型中的概念和二维表各部分的对应关系

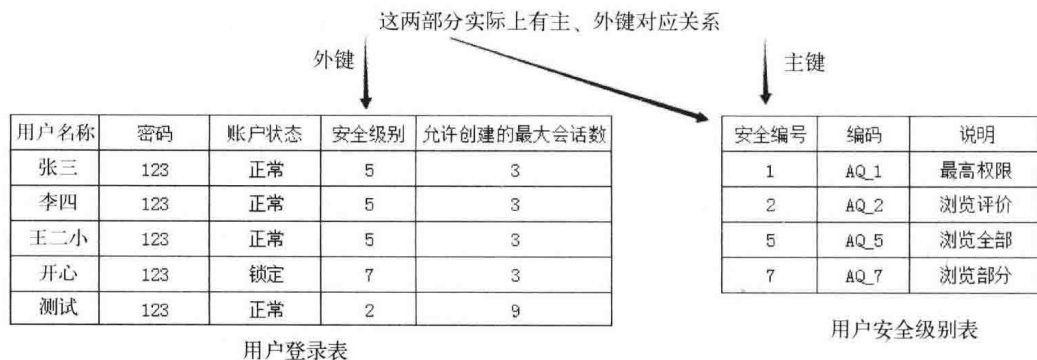


图 1-5 表间的参照完整性

在一个数据库中，要求表名唯一，同一个表中的字段名要求唯一，每个表至少有一个列。

1.1.3 数据库的设计

数据库设计是指在特定的环境中，设计结构和创建数据库的过程。数据库设计的目的是使之能够有效地存储数据和提供相应的服务。设计数据库分为需求分析、概念结构设计、逻辑结构设计、数据库物理结构设计、数据库实施、数据库运行与维护几个阶段。

在进行数据库设计时，开发者会经常接触概念结构设计这个阶段。该阶段会用到 E-R 图（Entity - Relationship Diagram）实体 - 联系图。E-R 图描绘的数据模型称为 E-R 模型。该模型对概念模型进行设计时通常会有以下个步骤：①建立局部概念模型（分 E-R 图）；②综合局部概念模型，完成全局概念模型（初步 E-R 图）；③消除冗余部分，成为最终的模型（基本 E-R 图）。

有关 E-R 图需要了解以下概念：

1. 实体

实体就是客观存在的对象，如学生王亮、渤海、太阳等。

2. 实体集

实体集是性质相同的同类实体的集合，如全体学生、大海等。

3. 属性

属性用来描述实体具有的特征，一个实体可以被多个属性描述。例如，学生具有的属性

有姓名、学号、年龄、班级等。

4. 实体型

实体名与带有的属性共同构成了实体型。

5. 联系

数据对象的相互关系称为联系。联系分为一对一关系 (1:1)，一对多 (1:N) 关系，多对多 (M:N) 关系三种方式。

(1) 一对一关系

一对一关系表示实体集 A 中的一个实体与实体集 B 中的一个实体有一对一的对应关系，反过来也如此。例如，一个学生只能做一个班级的班长，一个班级只能有一个班长；则班级和班长是一对一的关系。

(2) 一对多关系

一对多关系表示实体集 A 中的一个实体与实体集 B 中的多个实体相对应，而实体集 B 中的一个实体最多只能对应 A 中的一个实体。例如，一个学生可以担任多个学科的科代表，但每门课程只能有一个科代表。

(3) 多对多关系

多对多关系表示实体集 A 中的一个实体与实体集 B 中的多个实体相对应，同时实体集 B 中的一个实体也和实体集 A 中的多个实体相对应。例如，每个学生可以学习多门课程，每门课程允许多个学生来学习。

E-R 图中会用矩形框代表实体，矩形框内写实体名；用菱形框表示实体间的联系，菱形框内写联系名；用椭圆形表示实体的属性，并用直线将实体（或关系）与其属性连接起来。

【例 1-1】利用 E-R 图设计简单的学生选课系统数据库。

按照题目要求，涉及的实体集有学生、专业课程和教师。假设学生可以学习多门专业课程，一门专业课程可以被多个学生学习，则它们之间是多对多的关系；一个教师能教多门专业课程，一门专业课程允许由多个教师教授，则它们之间是多对多的关系。根据题目要求，步骤如下所示。

(1) 学生和专业课程之间的 E-R 图

对于学生实体集中的实体来说，每个学生都具有学号、系代码、专业代码、年级、姓名、性别、出生日期、入学时间等属性。课程实体集中的每一门专业课程都具有课程号、课程名、开课专业、学分、上课时间、上课地点等属性。为实体集选择其中一部分属性，并创建学生和专业课程的 E-R 图，如图 1-6 所示。

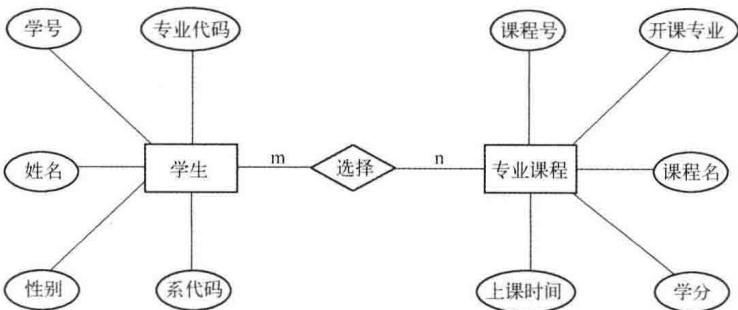


图 1-6 学生和专业课程的 E-R 图

(2) 教师和专业课程之间的 E - R 图

对于教师实体集中的实体来说，每个教师都具有教师编号、系代码、职称、姓名、性别、学历等属性。为实体集选择其中一部分属性，并创建教师和专业课程的 E - R 图，如图 1-7 所示。

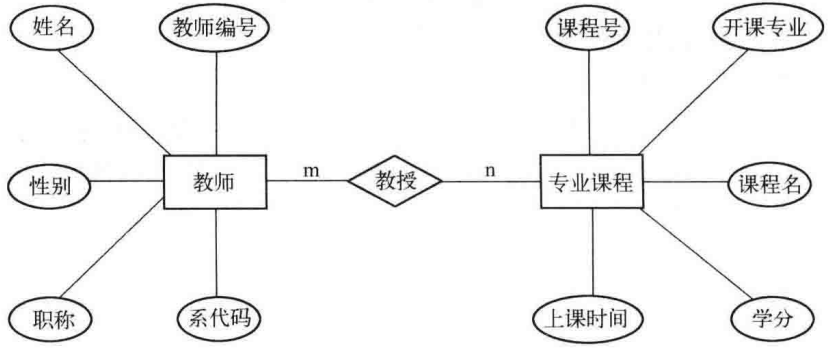


图 1-7 教师和专业课程的 E - R 图

(3) 教师、专业课程和学生之间的 E - R 图

整理前两步的局部 E - R 图，合成一个整体，如图 1-8 所示。

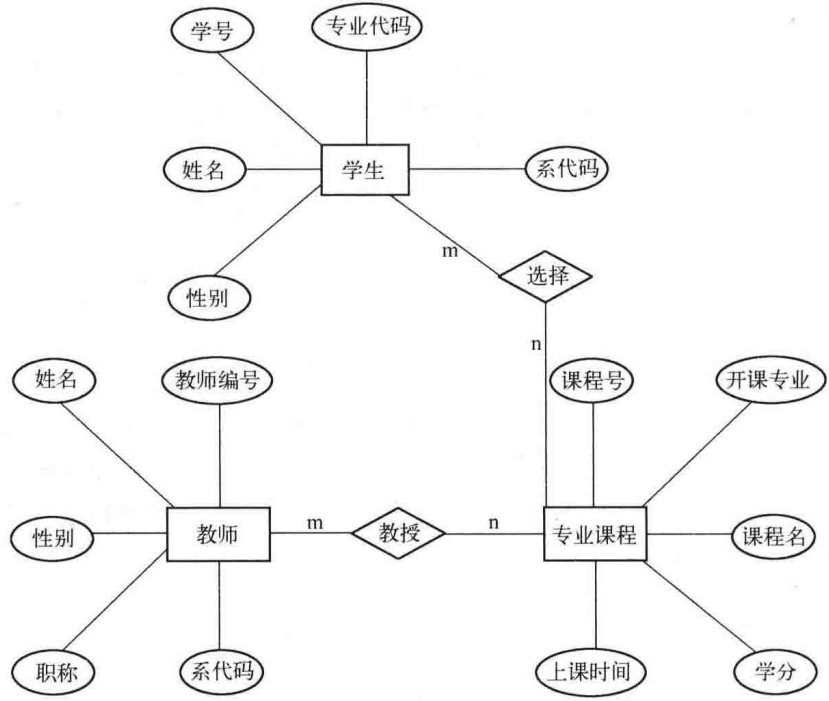


图 1-8 选课整体 E - R 图

图 1-8 是一个选课系统的基础核心内容，其中的属性和模块只给出了一部分。这样做的主要目的是让读者对 E - R 图有一个直观的认识，能够快速上手操作。

1.2 了解 SQL Server

SQL Server 是由微软出品的一款功能全面的关系型数据库管理系统。它运行在 Windows 平台上，与操作系统有机结合，是目前的主流数据库之一。它支持结构化查询语言，并提供了 In-Memory OLTP、BI 和混合云搭建功能。当前 SQL Server 的最新版数据库是 2016 版。考虑到实用性，本书将以 2014 版为基础，向读者介绍常用的数据库知识。

1.2.1 SQL Server 的发展历程

早期的数据库使用网状数据模型和层次数据模型来实现对数据的管理，检索其中的信息需要由了解数据本身结构的专业的编程人员来完成，因此它们并不算是理想的数据库模型。1970 年，IBM 公司的 E. F. Codd 发表了论文《大型共享数据库的关系数据模型》(A Relational Model of Data for Large Shared Data Banks)，其中首先提出了关系数据模型，并建议使用非过程语言来访问数据。此后 Codd 又提出了关系代数和关系演算的概念、函数依赖概念以及关系的第一、第二、第三范式，这为结构化查询语言奠定了基础，也为关系数据库系统奠定了理论基础。除此之外，Codd 也是最早提出了 OLAP 概念的人。

关系模型理论被提出后，受到了业界的广泛重视，而基于层次模型和网状模型的数据库产品则快速衰落，同时各大数据库厂商也推出了自己的关系型数据库产品。20 世纪 80 年代，微软和 Sybase 合作，并获得了 Ashton-Tate 支持，将 Sybase 开发的数据库纳入 OS/2 操作系统中，第一个名义上由微软开发的数据库 Ashton-Tate/Microsoft SQL Server 1.0 于 1989 年上市。此后，微软在 1990 年出品了 Microsoft SQL Server 1.1。

1992 年，微软和 Sybase 共同出品了 SQL Server 4.2。最初以 16 位开发，而后，由于 32 位操作系统开始盛行，于是微软用 Win32 API 重新编写数据库核心代码，并于 1993 年发布 SQL Server for Windows NT (4.2)。它是第一款应用在 Windows NT 上的 SQL Server，这标志着 Microsoft SQL Server 的正式诞生。

1994 年 4 月，微软向 Sybase 购买了有关 Windows 平台上 SQL Server 的代码版权，从此获取了对 SQL Server 的完全控制权，并于 1995 年发布了 SQL Server 6.0。该版本对微软来说具有里程碑的意义，因为这是完全由微软自行开发的产品，标志着微软数据库的研发能力得到了市场认可。在 1996 年，微软发布了 SQL Server 6.5。

在 20 世纪 90 年代，数据库竞争十分激烈。为了能让 SQL Server 有更好的兼容性以及允许其在便携式计算机中工作，微软对数据库核心重新编写，并在 1998 年底正式推出了 SQL Server 7.0。SQL Server 7.0 适用于企业应用。在市场竞争的压力下，微软再次升级数据库产品，并在 2000 年 8 月发布了 SQL Server 2000。

SQL Server 2000 是一个非常优秀的版本，其具有良好的数据处理能力、灵活的数据管理和分析能力以及便捷的操作性，很多商务网站、企业信息平台都使用了该产品。当时能与其竞争的产品只有 Oracle 和 IBM DB2，可谓三大数据库之一。

2005 年 11 月，SQL Server 2005 发布。该版本有了较大的变动，如使用 SQL Server Management Studio 取代了 SQL Server Enterprise Manager；增加了新的 Transact-SQL 命令；增强