



工业和信息化
人才培养规划教材

Industry And Information
Technology Training
Planning Materials

高职高专计算机系列

MySQL 数据库 任务驱动式教程

MySQL Database

石坤泉 汤双霞 王鸿铭 © 编著

- + 可作为**等级考试**辅导教材
- + 小型关系型数据库系统，**简单易学**，适合作为**Web 数据库**开发课程
- + 基于建构主义的**任务驱动式写法**，体现“**做中学**”“**学中做**”的思想



人民邮电出版社
POSTS & TELECOM PRESS



工业和信息化
人才培养规划教材

Industry And Information
Technology Training
Planning Materials

MySQL 数据库 任务驱动式教程

MySQL Database

石坤泉 汤双霞 王鸿铭 © 编著

人民邮电出版社
北京

图书在版编目 (C I P) 数据

MySQL数据库任务驱动式教程 / 石坤泉, 汤双霞, 王
鸿铭编著. — 北京: 人民邮电出版社, 2014.9

工业和信息化人才培养规划教材. 高职高专计算机系
列

ISBN 978-7-115-36271-1

I. ①M… II. ①石… ②汤… ③王… 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175053号

内 容 提 要

本书根据《全国计算机等级考试二级 MySQL 数据库程序设计考试大纲》编写而成, 主要讲述数据库设计的基本原理和基本方法、MySQL 的语言基础及其应用。全书共 11 个项目 24 个任务, 包括认识数据库, MySQL 实训环境配置, MySQL 字符集与数据类型, 建库、建表与数据表管理、数据查询, 数据处理与视图, 创建和使用程序, 数据库安全与性能优化, PHP 语言基础及应用, 访问 MySQL 数据库, PhpMyAdmin 操作数据库及 PHP+MySQL 综合项目开发。此外, 全书还配有习题和等级考试模拟题, 便于学习者巩固理论知识, 也可以作为学习者参加二级 MySQL 数据库程序设计考试的参考资料。

本书可以作为高职高专学生的数据库教材, 也可作为 MySQL 二级考试的参考教材, 以及供数据库开发人员使用的实用参考书。

-
- ◆ 编 著 石坤泉 汤双霞 王鸿铭
责任编辑 王 威
责任印制 焦志炜
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京艺辉印刷有限公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 18 2014 年 9 月第 1 版
字数: 472 千字 2014 年 9 月北京第 1 次印刷
-

定价: 42.00 元

读者服务热线: (010) 81055256 印装质量热线: (010) 81055316
反盗版热线: (010) 81055315

前言 PREFACE

本书的编写基于建构主义学习理论。建构主义学习理论认为，学习活动是一个以学习者为中心，学习者在个人原有知识经验基础上，在一定的社会文化中，主动地接受知识、积极地建构知识的过程。基于这个理论，本书采用项目（任务）驱动，项目（课内）—任务—知识—项目（课外）的编写模式，课内和课外项目“双项目并行”实施。希望通过实战项目驱动，让学习者主动建构知识和技能。

本书充分体现职业教育“理论够用”的原则，体现“做中学”、“学中做”CDIO 工程教育的思想。精心设计串起全课程的项目，项目是比较综合的技能组合，又分解成若干任务，任务具体包括“任务背景”、“任务要求”、“任务分解”、“分析点拨”等，每个任务结束后安排“项目实践”。“任务背景”用真实的案例或问题引入，调动学习者学习兴趣，让学习者有跃跃欲试的感觉。“任务要求”明确要学什么、学到什么程度，让学习者明确任务目标。然后进行“任务分解”，在任务安排上，力求做到循序渐进，由浅入深，层层递进，这是本书编写和教学的精髓部分。在“分析讨论”中对一些重点的 SQL 语句、对涉及的知识点和注意事项进行阐释说明，可以说是一个归纳总结和提升的部分。每个任务结束时安排一个综合性的“项目实践”，有效考察学习者对知识和技能的掌握程度和拓展应用能力。总之，本书将课程知识融入一个个项目（任务）的提出、分析、解决实施过程，使学习者掌握课程必须的知识和技能，然后将所有知识和技能应用于项目实践，真正达到学以致用目的。

MySQL 是小型关系型数据库系统，相比其他的数据库，有语言简单、易学的特点。常用的 SQL 语句主要有 SELECT、INSERT、CREATE、UPDATE、DROP 和 DELETE 等。MySQL 支持多平台操作系统（Linux、Windows、Mac OS 等），MySQL 也为 Java、C++、C# 和 PHP 等多种编程语言提供了 API，典型的组合如 LAMP（Linux+Apache+MySQL+PHP）、WAMP（Windows 平台下的 Apache+MySQL+PHP）。MySQL 支持多国语言，MySQL 的字符集有服务器级、数据库级、表级和连接级等层级，要注意各级字符集的统一，不然会出现乱码或字符集混杂的问题。

本书采取命令行与界面工具相结合的教学方式。MySQL 客户端、WAMP 为学习者提供了友好直观的学习环境。WAMP 是一款优秀的 WAMP 集成软件，可以轻松地搭建好 Apache+MySQL+PHP5 环境，学习和管理数据库非常方便。

为方便教学，本书提供配套的课件、课程源程序文件、教学示例数据库、项目实践数据库，以及 PHP+MySQL 开发的应用系统文件。有需要的读者可登录人民邮电出版社教学服务与资源网（www.ptpedu.com.cn）免费下载。

本书由广州番禺职业技术学院石坤泉、汤双霞和平顶山工业职业技术学院王鸿铭编著，其中，任务 1~任务 19 由石坤泉编写，任务 20 与任务 24 由汤双霞编写，任务 21~任务 23 由王鸿铭编写。

由于编者水平有限，书中难免出现错误与不足，希望读者指正与谅解。

编者

2014 年 7 月

目 录 CONTENTS

项目一 认识数据库 1

任务 1 认识数据库	1	任务 2 认识 MySQL	15
1.1 了解数据库的基本应用	1	2.1 认识 SQL 语言	15
1.2 了解数据库的几个概念	2	2.2 MySQL 概述	16
1.3 认识关系型数据库	3	2.3 认识 MySQL 5.5	17
1.4 关系数据库设计	4	2.4 MySQL 的管理工具	18
1.5 数据库应用系统体系结构	9	2.5 初识 MySQL 数据类型	19
1.6 认识一个真实的关系型数据库	11	2.6 初识 MySQL 的基本语句	20

项目二 MySQL 实训环境配置 22

任务 3 Windows 环境下 MySQL 的安装与配置	22	3.4 远程访问 MySQL 服务器	30
3.1 MySQL 服务器的安装与配置	22	任务 4 安装配置 WAMP Server 2.2	32
3.2 更改 MySQL 的配置	27	4.1 认识 WAMP Server 2.2	32
3.3 连接 MySQL 本地服务	28	4.2 安装 WAMP Server 2.2	32
		4.3 配置 WAMP Server 2.2	34

项目三 MySQL 字符集与数据类型 37

任务 5 认识和设置 MySQL 字符集	37	6.2 浮点数类型和定点数类型	45
5.1 认识字符集和校对原则	37	6.3 CHAR 类型和 VARCHAR 类型	45
5.2 MySQL 5.5 支持的字符集和校对原则	38	6.4 TEXT 类型和 BLOB 类型	45
5.3 确定字符集和校对原则	40	6.5 BINARY 和 VARBINARY	46
5.4 使用 MySQL 字符集时的建议	42	6.6 时间和日期类型	46
任务 6 MySQL 数据类型	44	6.7 ENUM 类型和 SET 类型	47
6.1 整数类型	44	6.8 如何选择数据类型	47
		6.9 数据类型的附加属性	48

项目四 建库、建表与数据库管理 50

任务 7 建立数据库和表	50	8.3 索引的查看	71
7.1 创建与管理数据库	50	8.4 索引的删除	71
7.2 创建与管理表	52	任务 9 数据约束和参照完整性	74
7.3 表数据操作	58	9.1 PRIMARY KEY 约束	74
任务 8 建立和管理索引	66	9.2 UNIQUE 约束	76
8.1 认识索引	66	9.3 FOREIGN KEY 参照完整性约束	77
8.2 索引的建立	68	9.4 CHECK 约束	79

项目五 数据查询、数据处理与视图 81

任务 10	数据库的查询	81	11.2	认识和使用函数	115
10.1	了解 SELECT 语法结构	81	任务 12	创建和使用视图	132
10.2	认识基本子句	82	12.1	认识视图	132
10.3	使用聚合函数进行查询统计	92	12.2	视图的特性	132
10.4	多表连接查询	93	12.3	创建视图	133
10.5	嵌套查询	96	12.4	查看视图	135
10.6	联合查询	101	12.5	使用视图	136
任务 11	MySQL 运算符和函数	105	12.6	修改视图	138
11.1	认识和使用运算符	105	12.7	删除视图	138

项目六 创建和使用程序 140

任务 13	建立和使用存储过程	140	14.4	查看存储函数	156
13.1	认识存储过程	140	14.5	修改存储函数	156
13.2	创建基本的存储过程	141	14.6	删除存储函数	157
13.3	执行存储过程	144	任务 15	建立和使用触发器	159
13.4	创建带变量的存储过程	145	15.1	认识触发器	159
13.5	创建带有流程控制语句的存储过程	147	15.2	创建触发器	160
13.6	在存储过程中调用其他存储过程	150	15.3	查看触发器	163
13.7	修改存储过程	150	15.4	删除触发器	164
13.8	查看存储过程的定义	151	任务 16	建立和使用事件	165
13.9	删除存储过程	151	16.1	认识事件	165
任务 14	建立和使用存储函数	153	16.2	创建事件	166
14.1	认识存储函数	153	16.3	查看事件	168
14.2	创建存储函数	153	16.4	修改事件	169
14.3	调用存储函数	155	16.5	删除事件	170

项目七 数据库安全与性能优化 171

任务 17	用户与权限	171	18.2	用 LOAD DATA INFILE 恢复表数据	184
17.1	创建用户账户	171	18.3	用 mysqldump 备份与恢复	184
17.2	授予用户权限	175	18.4	用 mysqlimport 恢复表数据	188
17.3	用 REVOKE 收回权限	179	18.5	用 SOURCE 恢复表和数据库	188
17.4	权限转移	179	18.6	直接复制移植	189
17.5	权限限制	179	18.7	用日志备份	189
17.6	初始 MySQL 账户安全	180	任务 19	数据库性能优化	191
17.7	密码安全	181	19.1	优化 MySQL 服务器	191
任务 18	数据库备份与恢复	183	19.2	优化表结构设计和数据操作	193
18.1	用 SELECT INTO OUTFILE 备份表数据	183	19.3	优化查询	198
			19.4	优化性能的其他方面	205

项目八 PHP 语言基础及应用 207

任务 20	PHP 初识与应用	207	20.5	PHP 操作 MySQL 数据库	219
20.1	PHP 技术基础	207	20.6	PHP 备份与还原 MySQL 数据库	221
20.2	PHP 的数据类型	209	20.7	应用实践：基于文本的 简易留言系统	222
20.3	PHP 数据处理	211			
20.4	PHP 连接 MySQL 数据库	218			

项目九 访问 MySQL 数据库 226

任务 21	Java 访问 MySQL	226	任务 22	C#访问 MySQL 数据库	231
21.1	Java 连接 MySQL 数据库	226	22.1	C#连接 MySQL 数据库	231
21.2	Java 操纵 MySQL 数据库	228	22.2	C#操作 MySQL 数据库	233
21.3	Java 备份 MySQL 数据库	229	22.3	C#备份与还原 MySQL 数据库	234
21.4	Java 还原 MySQL 数据库	229			

项目十 PhpMyAdmin 操作数据库 235

任务 23	phpMyAdmin 操作数据库	235	23.7	创建视图	242
23.1	创建与管理数据库	235	23.8	创建和使用程序	242
23.2	创建与管理表	236	23.9	创建和使用触发器	245
23.3	字符集设置	238	23.10	创建事件和使用事件	246
23.4	表数据操作	239	23.11	用户与权限管理	248
23.5	索引与参照完整性约束	240	23.12	备份与恢复数据库	250
23.6	使用查询	241			

项目十一 PHP+MySQL 数据库开发 252

任务 24	学生学习论坛系统	252	24.4	总体及界面设计	253
24.1	系统规划	252	24.5	数据库设计	255
24.2	系统功能	252	24.6	界面设计	256
24.3	详细功能	253	24.7	代码设计	259

附录 A MySQL 常用语句 271

附录 B 存储引擎 273

附录 C 全国计算机等级考试二级 MySQL 数据库程序设计考试大纲 276

附录 D 全国计算机等级考试二级 MySQL 数据库程序设计考试样题 278

任务 1 认识数据库

【任务背景】

学习数据库，一般都要先从数据库基本原理开始学起。对于初学者，往往会感觉原理部分特别枯燥乏味，面对很多抽象难懂的概念和理论知识，如数据模型、实体、属性、联系、E-R 图、关系模式和数据建模等，显得很无奈。但是，这些内容恰恰又是数据库开发人员必须具备的基本知识。那么，该如何组织好教学内容？怎样组织教学？怎样才能使讲授变得通俗易懂？该如何培养学生对课程学习的兴趣？带着很多的问题和期盼，让我们一起努力，走进数据库的世界去探究吧！

【任务要求】

本任务直接从数据库的基本应用开始，让学生感受一下身边的数据库应用。在具体的学习情景中，让学生了解数据库的基本概念；认识关系模型和关系型数据库；认识实体和属性；认识 C/S、B/S 模式架构；学习数据库的概念结构设计和逻辑设计方法；学习关系模式的规范化；绘制 E-R 图，建立数据库概念模型；将 E-R 图转换成关系模式进行数据模型的建立。

【任务分解】

1.1 了解数据库的基本应用

数据库管理系统应用非常广泛，可以说应用在各行各业。不管是家庭、公司或大型企业，还是政府部门，都需要使用数据库来存储数据信息。对软件而言，无论是 C/S 还是 B/S 架构的软件，只要涉及存储大量数据，一般都需要数据库支撑。传统数据库很大一部分用于商务领域，如证券行业、银行、销售部门、医院、公司或企业单位，以及国家政府部门、国防军工领域和科技发展领域等。随着信息时代的发展，数据库也相应地产生了一些新的应用领域。

主要表现在多媒体数据库、移动数据库、空间数据库、信息检索系统、分布式信息检索和专家决策系统等几个方面。

下面介绍几种数据库管理系统在实际应用中的案例。

1. 留言板

留言板是互联网上最常见的一项功能，留言板的数据库存储的数据不大。其数据库大致分为下面几个版块。用户注册和登录功能版块：后台数据库需要存放用户的注册信息和在线状态信息；管理用户发帖版块：后台数据库需要存放帖子相关信息，如帖子内容、标题等；管理论坛版块：后台数据库需要存放各个版块信息，如版主、版块名称和帖子数等。

2. 进销存管理系统

进销存管理系统是有效辅助企业解决业务管理、分销管理、存货管理、营销计划的执行和监控、统计信息的收集等方面的业务的管理系统。要实现的一般功能包括：批发销售零售销售、供货商往来账务管理、客户往来账务管理、提供业务员和员工管理、提供 POS 端销售管理、支持财务管理功能和支持库存盘点功能等。

3. ERP 系统

ERP 是 Enterprise Resource Planning（企业资源计划）的简称。ERP 系统是建立在信息技术基础上，以系统化的管理思想，为企业决策层及员工提供决策运行手段的管理平台。建立企业的信息管理系统，支持大量原始数据的查询、汇总。借助计算机的运算能力及系统对客户订单、在库物料和产品构成的管理能力，实现依据客户订单，按照产品结构清单展开并计算物料需求计划，实现减少库存、优化库存的管理目标等。

4. 图书管理系统

图书管理系统概念结构主要由 4 大部分组成，即信息源、信息处理器、信息用户和信息管理者。软件系统通常运行在 C/S 或 B/S 模式，包括图书的采访、编目、流通、查询、期刊管理、系统管理、字典管理和 Web 检索与发布等图书借阅管理系统子系统。

1.2 了解数据库的几个概念

1. 数据（Data）

数据实际上就是描述事物的符号记录。文本、数字、时间日期、图片、音频和视频等都是数据。数据的特点：有一定的结构，有型与值之分，如整型、字符型、文本型等。而数据的值给出了符合定型的值，如 INT(15) 表示 15 位整型数据，VARCHAR(8) 表示 8 个长度的字符。

2. 数据库（DataBase, DB）

数据库是长期储存在计算机内的、有组织的、可共享的数据集合。数据库中的数据按一定的数据模型组织、描述和储存，具有较小的冗余度、较高的数据独立性和易扩展性，并可为各种用户共享。

3. 数据库管理系统（DBMS）

数据库管理系统是一个负责对数据库进行数据组织、数据操纵、维护、控制以及数据保护和数据服务等软件系统，是数据库的核心。能够让用户定义、创建和维护数据库以

及控制对数据的访问。它对数据库进行统一的管理和控制,以保证数据库的安全性和完整性。用户通过 DBMS 访问数据库中的数据,数据库管理员也通过 DBMS 进行数据库的维护工作。

数据库管理系统的主要功能如下。

- (1) 数据模式定义:即为数据库构建其数据框架。
 - (2) 数据存取的物理构建:为数据模式的物理存取与构建提供有效的存取方法与手段。
 - (3) 数据操纵:为用户使用数据库的数据提供方便,如查询、插入、修改和删除等以及简单的算术运算及统计。
 - (4) 数据的完整性、安全性定义与检查。
 - (5) 数据库的并发控制与故障恢复。
 - (6) 数据的服务:如复制、转存、重组、性能监测和分析等。
- 数据库管理系统一般要提供以下的数据语言。

- (1) 数据定义语言:负责数据的模式定义与数据的物理存取构建。
- (2) 数据操纵语言:负责数据的操纵,如数据查询与增、删、改数据等。
- (3) 数据控制语言:负责数据完整性、安全性的定义与检查以及并发控制、故障恢复等。

常用的数据库管理系统有 MySQL、MS SQL Server、Oracle、DB2、Access 和 Sybase 等。

MySQL 是一个小型关系型数据库管理系统,最初由 MySQL AB 公司开发与发布,后来被 sun 公司收购,从 2009 年开始,MySQL 又成为了 Oracle 公司旗下的另一个数据库项目。MySQL 被广泛地应用在 Internet 上的中、小型网站中。本书讲述的是最新版本 MySQL 5.5。

1.3 认识关系型数据库

1. 认识数据模型

数据模型由 3 部分组成,即模型结构、数据操作和完整性规则。DBMS 所支持的数据模型分为 3 种:层次模型、网状模型和关系模型。

用树形结构表示实体及其之间的联系模型称为层次模型。该模型的实际存储数据由链接指针来体现联系。特点:有且仅有一个结点无父结点,此结点即为根结点;其他结点有且仅有一个父结点。适合用来表示一对多的联系。

用网状结构表示实体及其之间的联系模型称为网状模型。允许结点有多于一个的父结点,可以有一个以上的结点无父结点。适合于表示多对多的联系。

层次模型和网状模型本质上都是一样的。存在的缺陷:难以实现系统扩充,插入或删除数据时,涉及大量链接指针的调整。

在关系模型中,一个关系就是一张二维表,通常将一个没有重复行、重复列的二维表看成一个关系,每个关系都有一个关系名。二维表的每一行在关系中称为元组(记录),二维表的每一列在关系中称为属性(字段),每个属性都有一个属性名,属性值则是各元组属性的取值。

2. 认识关系数据库

关系数据库是创建在关系模型基础上的数据库。关系数据库是以二维表来存储数据库的数据的。

例如, YSGL 数据库的 EMPLOYEES 表中的数据, 如图 1.1 所示, 有 E_ID、E_name、sex、Professional、education、Political、birth、marry、Gz_time、D_id 和 bz 等 11 个字段, 分别代表员工的员工编号、员工姓名、性别、职称、政治面貌、学历、出生日期、婚姻状态、参加工作时间、部门 ID 和编制情况等信息, 这反映了表的结构。

E_ID	E_name	sex	Professional	education	Political	birth	marry	Gz_time	D_id	bz
100100	李明	男	副教授	硕士	党员	1967-02-01	否	1989-09-01	B001	是
100101	李小光	男	讲师	本科	党员	1985-03-01	否	1990-10-02	B001	是
100102	张伟健	男	教授	本科	党员	1965-05-06	是	1987-07-08	B003	是
100103	石小华	女	教授	硕士	党员	1978-06-07	是	1992-01-01	A001	是
100104	黄莉	女	助讲	硕士	群众	1986-03-04	否	2001-05-06	A002	是
100105	余明平	男	教授	硕士	党员	1960-05-12	是	1983-04-05	B001	是
100106	苏小明	男	教授	硕士	群众	1956-04-13	是	1983-04-03	B003	是
100107	汤光明	男	教授	硕士	群众	1983-01-04	是	2007-12-06	A004	是
100108	谢建设	男	副教授	博士	民进	1987-02-08	否	2006-08-15	A005	是
100109	胡晓群	女	讲师	博士	党员	1990-09-28	否	2012-05-23	B005	是
100330	李正中	男	副教授	硕士	群众	1978-10-29	否	1995-10-01	B002	是
100331	王君君	女	教授	博士	党员	1956-11-18	否	1978-12-07	B003	是
100332	赵剑	男	讲师	博士	党员	1967-12-25	否	1989-04-08	B005	是
100333	欧阳	女	副教授	博士	民进	1966-03-08	否	1988-02-09	B003	是
200100	李明义	男	讲师	硕士	党员	1965-04-17	否	1987-08-10	B005	是
200101	孙美灵	男	讲师	硕士	党员	1977-05-16	否	1998-09-11	B003	是
200102	王世明	男	副教授	硕士	党员	1976-03-21	否	1996-07-21	A004	是
200103	张平娜	女	讲师	硕士	党员	1989-04-23	否	2008-06-22	A001	是
200104	李美丽	女	讲师	博士	九三学社	1982-12-01	是	0000-00-00	B001	是
200105	苏珍珍	女	副教授	本科	党员	1966-08-11	是	1985-08-01	B003	是
200220	张白燕	女	讲师	博士	群众	1987-05-01	是	2011-09-02	A001	是
200221	李青青	女	副教授	本科	群众	1964-10-06	是	1989-04-03	A001	是

图 1.1 EMPLOYEES 表数据

关系模式是对关系的描述, 主要描述关系由哪些属性构成, 即描述了表的结构。关系模式的格式为关系表名(字段 1, 字段 2, 字段 3, ..., 字段 n)。因此, EMPLOYEES 表的关系模式可以表示为

```
EMPLOYEES(E_ID, E_name, sex, Professional, education, Political, birth, marry, Gz_time, D_id, bz)
```

在关系表中, 通常指定某个字段或字段的组合的值来唯一地表示对应的记录, 我们把这个字段或字段的组合称为主码(也叫主键或关键字, 下同)。如在 EMPLOYEES 表中, 一个雇员一个编号, 其值是唯一的, 可以用雇员编号唯一表示一位雇员, 因此, 把 E_ID 指定为 EMPLOYEES 表的主码。

表中的每一行是一条记录, 记录一位雇员的相关信息。例如, 第一位员工的信息分别是: 100100, 李明, 男, 副教授, 硕士, 党员, 1967-02-01, 否, 1989-9-1, B001, 是。

1.4 关系数据库设计

数据库设计一般要经过需求分析、概念结构设计、逻辑设计、物理设计、数据库实施和数据运行等几个阶段。其中, 概念结构设计阶段要对需求进行综合、归纳与抽象, 形成一个

独立于具体 DBMS 的概念模型（用 E-R 图表示），逻辑设计阶段是将概念结构转换为某个 DBMS 所支持的数据模型（如关系模式），并对其进行优化。下面就概念结构设计阶段和逻辑设计阶段要做的事分述如下。主要是认识实体与属性，绘制 E-R 图，并将 E-R 图转换为关系模式。

1.4.1 实体、属性、联系

客观存在并相互区别的事物称为实体。实体是一个抽象名词，是指一个独立的事物个体，自然界的一切具体存在的事物都可以看作为一个实体。一个人是一个实体，一个组织也可以看作为一个实体。在学校里，学校是一个实体，院系是一个实体，学生也是一个实体。要描述一位学生，通常用他的学号、姓名和年龄等特征信息项来表示。那么，学号、姓名和年龄就是学生的属性，不同学生属性不同。例如，课程是一个实体，课程的属性通常是课程编号、课程名称和学分等；教师是一个实体，教师的属性可以是教师编号、姓名、性别和职称等；院系单位也是一个实体，其属性可以用单位编号、单位名称和电话等来表示。

实体与实体之间往往存在关系，这种关系也叫作“联系”。实体与实体的联系通常有 3 种：一对一（1:1）关系、一对多（1:M）关系和多对多（M:N）关系。例如，学校与校长是一对一关系，一个学校只能有一个校长，一个校长只能属于一个学校；单位与教师的关系是一对多关系，一个教师只能属于一个单位，一个单位有多个教师；教师与课程的关系是多对多关系，一个教师可以讲授多门课，一门课可以由多个教师讲授。

1.4.2 绘制 E-R 图

可以用图来更加直观地描述实体和实体的联系，这样的图叫作 E-R 图（Entity-Relationship Approach）。一般情况下，用矩形表示实体，用椭圆表示实体的属性，用菱形表示实体型之间的联系，并用无向直线将其与相应的实体连接起来。

在建立 E-R 图时，特别是对于多个实体的比较复杂的 E-R 图，可以首先设计局部 E-R 模式，然后把各局部 E-R 模式综合成一个全局的 E-R 模式，最后对全局 E-R 模式进行优化，得到最终的 E-R 模式。

例如，在 XSGL 数据库中，有教师、课程、学生和院系 4 个实体。首先，绘制教师与课程的 E-R 图，如图 1.2 所示。

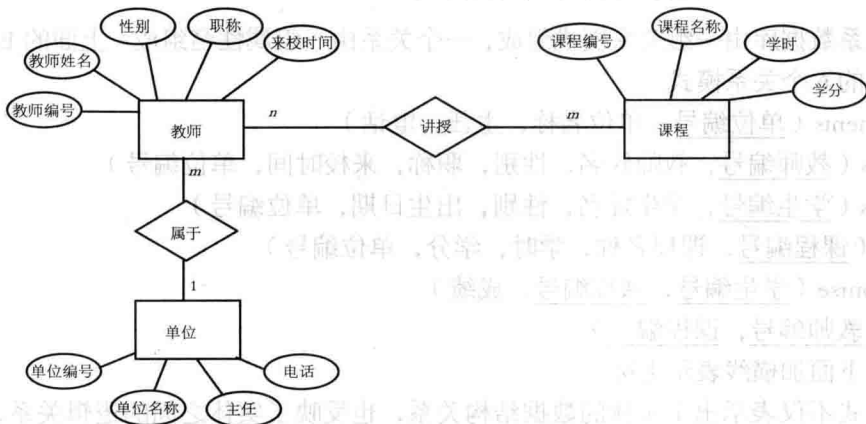


图 1.2 教师与课程的 E-R 图

然后，绘制学生与课程的 E-R 图，如图 1.3 所示。

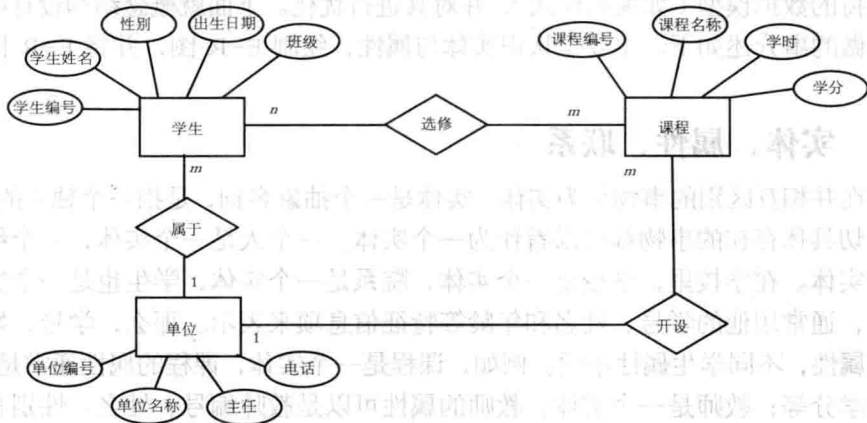


图 1.3 学生与课程的 E-R 图

最后，绘制全局 E-R 图，如图 1.4 所示。

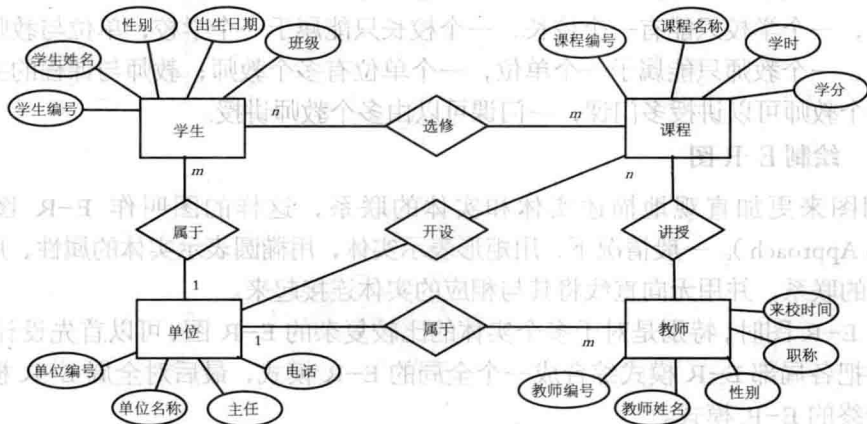


图 1.4 全局 E-R 图

1.4.3 从 E-R 图转换为关系模式

一个关系数据库由一组关系模式组成，一个关系由一组属性名组成。上面的 E-R 图可以转换为如下的 6 个关系模式。

departments (单位编号, 单位名称, 主任, 电话)

teachers (教师编号, 教师姓名, 性别, 职称, 来校时间, 单位编号)

students (学生编号, 学生姓名, 性别, 出生日期, 单位编号)

course (课程编号, 课程名称, 学时, 学分, 单位编号)

selectcourse (学生编号, 课程编号, 成绩)

teach (教师编号, 课程编号)

在属性下面加横线表示主键。

关系模式不仅表示出了实体的数据结构关系，也反映了实体之间的逻辑关系，具体描述如下。

院系与教师是一对多的关系，把关系模式 departments 的主键 (单位编号) 加入到 teachers

的关系模式中, 作为 teachers 的一个属性, 建立起院系与教师的联系;

学生与院系是一对多的关系, 把关系模式 departments 的主键(单位编号)加入到 students 的关系模式中, 作为 students 的一个属性, 建立起院系与学生的联系;

院系与课程的关系是一对多的关系, 把院系的主键加入到关系模式 course 中, 作为 course 中一个属性, 建立起院系与课程的关系;

教师与课程是多对多的关系, 要单独对应一个关系模式, 把关系模式 teachers 的主键和关系模式 course 主键一起加入到关系模式 teach, 作为 teach 的主键, 建立起教师与课程的关系;

学生与课程是多对多的关系, 要单独对应一个关系模式, 把关系模式 students 的主键和关系模式 course 的主键一起加入到关系模式 selectcourse, 作为 selectcourse 的主键, 建立起学生与课程的关系。

根据设计好的关系模式, 就可以创建数据表了。按照上面的关系模式, 可以创建 departments、teachers、course、students、teach 和 selectcourse 等 6 张表。

1.4.4 关系模式的规范化

构造数据库必须遵循一定的规则。在关系数据库中, 这种规则就是范式。关系按其规范化程度从低到高可分为 5 级范式, 分别称为 1NF、2NF、3NF (BCNF)、4NF 和 5NF。规范化程度较高者必是较低者的子集。满足最低要求的范式是第一范式 (1NF)。在第一范式的基础上进一步满足更多要求的称为第二范式 (2NF), 其余范式以此类推。一般说来, 数据库只需满足第三范式。下面, 举例介绍第一范式 (1NF)、第二范式 (2NF) 和第三范式 (3NF)。

1. 第一范式 (1NF)

在任何一个关系数据库中, 第一范式 (1NF) 是对关系模式的基本要求, 不满足第一范式 (1NF) 的数据库就不是关系数据库。

所谓第一范式 (1NF) 是指数据库表的每一列都是不可分割的基本数据项, 同一列中不能有多个值, 即实体中的某个属性不能有多个值或者不能有重复的属性。简而言之, 第一范式就是无重复的列。

例如, teachers (教师编号, 教师姓名, 性别, 职称), 每一列都是不可分割的最小数据项。表的每一行包含一位教师的信息, 满足 1NF。

反之, 假设有如下的关系模式: teachers (教师编号, 教师姓名, 性别, 职称, 联系方式), 由于联系方式有多种, 可以是手机、电话、QQ 等方式, 电话还可能有多, 例如家庭电话和办公电话, 可见“联系方式”并不是最小的数据项。因此, 该关系模式不符合 1NF, 可以把上面的关系模式改成:

teachers (教师号, 姓名, 性别, 职称, 手机, 家庭电话, 办公电话, QQ)

事实上, 并不是符合 1NF 的表就是规范合理的表。图 1.5 是一个学生信息表, 虽然每一列是不能再可分割的最小数据项, 但是, 在所在系部、专业列存在大量重复的数据, 因此, 这样的表也是不规范的。那么应该怎样地再进一步规范呢? 下面来学习 2NF 和 3NF。

2. 第二范式 (2NF)

第二范式 (2NF) 是在第一范式 (1NF) 的基础上建立起来的, 即满足第二范式 (2NF) 必须先满足第一范式 (1NF)。第二范式 (2NF) 要求如下。

(1) 表必须有一个主键, 要求数据库表中的每个记录必须可以被唯一地区分。为实现区分通常需要为表加上一个列, 以存储各个记录的唯一标识 (即主键)。

(2) 实体的属性完全依赖于主键, 而不能只依赖于主键的一部分(有时主键是由多个列组成的复合主键)。如果存在, 那么这个属性和主键的这一部分应该分离出来形成一个新的实体, 新实体与原实体之间是一对多的关系。

s_no	s_name	sex	birthday	department	project	address
学号	姓名	性别	出生日期	所在系部	专业	家庭地址
122001	张群	男	1990-02-05	金融学院	国际金融	文明路8号
122002	张平	男	1992-03-02	金融学院	国际金融	人民路9号
122003	余亮	男	1992-06-03	金融学院	国际金融	北京路188号
122004	李军	女	1993-02-01	金融学院	国际金融	东风路66号
122005	刘光明	男	1992-05-06	管理学院	人力资源	东风路110号
122006	叶明	女	1992-05-02	管理学院	人力资源	学院路89号
122007	张早	男	1992-03-04	管理学院	人力资源	人民路67号
122008	陈小春	男	1978-02-01	外语学院	应用外语	NULL
122009	李小雨	男	1990-03-08	信息工程	计算机网络	青山湖路
122010	吴天	男	1990-03-08	机电学院	电子技术	市南路1234号
123003	马志明	男	1992-06-02	信息工程	嵌入式技术	安西路10号
123004	吴文辉	男	1992-04-05	信息工程	嵌入式技术	学院路9号
123006	张东妹	女	1992-06-07	信息工程	网络技术	溁明路223号
123007	方莉	女	1992-07-08	信息工程	通讯技术	东风路6号
123008	刘想	女	1992-03-04	信息工程	通讯技术	中山路56号

图 1.5 数据表示例

总之, 第二范式就是要有一主键, 同时, 非主属性非部分依赖于主关键字。

例如, teachers (教师编号, 教师姓名, 性别, 职称) 中的属性“教师号”能唯一地标识教师。一个教师一个编号, 编号是唯一的。因此, 每个教师可以被唯一区分。其他非主键中的列“姓名”、“性别”、“职称”也完全依赖于主键。满足了第二范式。

反之, 如果想建立一个学生选修情况表, 关系模式如下。

Selectcourse (学生号, 课程号, 课程名, 学时, 学分, 成绩)

在这个关系中, 选择“学生号”和“课程号”作为复合主键。一个学生一门课只能有一个成绩, “成绩”是完全依赖于主键(学生号, 课程号)。然而, 课程名、学时、学分只能依赖于课程号(主键的一部分), 这样的关系模式不符合 2NF。

在实际应用中, 使用这样的关系模式会出现以下问题。

(1) 插入异常。学生号和课程号共同组成关键字, 要录入课程信息必须知道学生号这个关键字。对于新课, 由于还没有人选修, 就没办法录入课程信息。也就是说, 无法开出开课计划, 只能等有人选修后才能把课程信息录入。

(2) 数据冗余。假设同一门课有 50 个学生选修, 在录入学生成绩时, 课程名、学时、学分等信息就必须重复录入 49 次, 而且, 学分能否计入还要看成绩而定。

(3) 更新异常。如果调整了某课程的学时、学分, 相应记录的学时、学分值也要更新, 工作量非常大, 若录入不慎, 可能还会出现几个地方同一门课学时、学分不相同的错误。

(4) 删除异常。由于学生成绩记录与课程信息在同一个表中, 假设一批学生已经完成课程的选修, 这些选修记录就应该从数据库表中删除。但是, 原本要保存的课程及学时、学分信息也随之删除, 以后又要重新插入, 又会导致插入异常。

所以, Selectcourse 表不符合 2NF, 应该把它进行拆分, 将课程号、课程名、学时、学分从原关系中分离出来形成一个新的实体 course, 以解决数据冗余、更新、插入和删除的问题。

新的关系模式如下：

course (课程号, 课程名, 学时, 学分)

selectcourse (学生号, 课程号, 成绩)

3. 第三范式 (3NF)

满足第三范式 (3NF) 必须先满足第二范式 (2NF), 并且, 要消除传递函数依赖。数据表中如果不存在非关键字段对任一候选关键字段的传递函数依赖, 则符合第三范式。所谓传递函数依赖, 指的是如果存在 “ $A \rightarrow B \rightarrow C$ ” 的决定关系, 则 C 传递函数依赖于 A 。因此, 满足第三范式的数据库表应该不存在如下依赖关系: 关键字段 \rightarrow 非关键字段 $x \rightarrow$ 非关键字段 y 。

例如, teachers (教师号, 姓名, 性别, 系别, 系名, 系主任, 院系电话)

在这一关系中, 关键字教师号决定各个属性, 是主关键字。由于是单个关键字, 没有部分依赖的问题, 肯定是 2NF。但是这样的关系模式存在一个问题, 关系中存在传递依赖:

(教师号) \rightarrow (系别) \rightarrow (系名, 系主任, 院系电话)

即非主关键字段 (系名、系主任、院系电话) 函数依赖于候选关键字段 (系别), 并传递函数依赖于主码 (教师)。

在实际应用中, 这会造成系别、系名、系主任、院系电话等信息将重复存储, 会有大量的数据冗余, 就如图 1.5 所示的数据表。同时, 也会存在更新异常、插入异常和删除异常的情况。因此, 可以把原关系模式分拆为两个, 如下。

Teachers (教师号, 姓名, 性别, 系别)

departments (系别, 系名, 系主任, 院系电话)

新关系包括两个关系模式, 它们之间通过 teachers 中的外键 (系别) 进行联系, 需要时再进行自然联接, 恢复了原来的关系。

同理, 图 1.5 所示的数据表也可以分拆为两个。

```
Students(s_no, s_name, sex, birthday, address, d_ID)
```

```
Departments(d_ID, d_name, project)
```

4. 第二范式 (2NF) 和第三范式 (3NF) 的区别

第二范式 (2NF) 和第三范式 (3NF) 的概念很容易混淆, 区分它们的关键点如下。

(1) 2NF: 非主键列是否完全依赖于主键, 还是依赖于主键的一部分。

(2) 3NF: 非主键列是直接依赖于主键, 还是直接依赖于非主键列。

1.5 数据库应用系统体系结构

目前, 常用的以数据库为核心构成的应用系统多数采用了以下两种结构模式: 客户机/服务器 (Client/Server, C/S) 模式和三层客户机/服务器 (Browser/Server, B/S) 模式。

1.5.1 认识 C/S 模式数据库

C/S (Client/Server 或客户机/服务器模式) 结构, 即大家熟知的客户机/服务器结构。Client 和 Server 常常分别处在相距很远的两台计算机上, Client 程序的任务是将用户的要求提交给 Server 程序, 再将 Server 程序返回的结果以特定的形式显示给用户; Server 程序的任务是接收客户程序提出的服务请求, 进行相应的处理, 再将结果返回给客户程序。C/S 模式数据库应用系统的开发工具通常是 Visual Basic (VB)、Visual C#、Visual C++/NET、Delphi、Power

Builder (PB) 等。图 1.6 是三层 C/S 模式应用系统结构图。

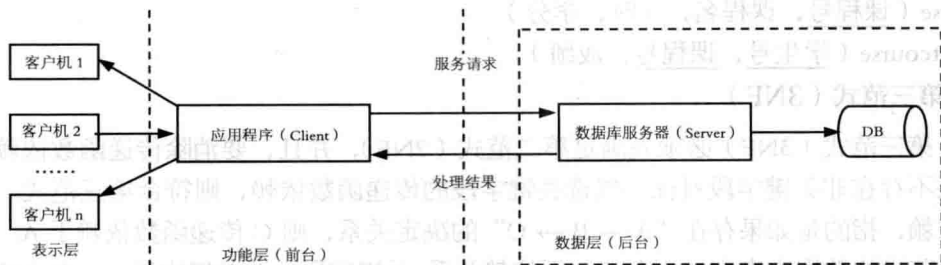


图 1.6 三层 C/S 模式应用系统

1.5.2 认识 B/S 模式数据库

随着 Internet 和 WWW 的流行，以往的主机/终端和 C/S 结构都无法满足当前的全球网络开放、互连、信息随处可见和信息共享的新要求，于是就出现了 B/S 模式。如图 1.7 所示。



图 1.7 B/S 模式应用系统

B/S (Browser/Server 结构) 模式数据库是采用三层客户机/服务器模式，即浏览器和服务器的结构。用户工作界面是通过 WWW 浏览器来实现，主要事务逻辑在服务器端 (Server) 实现，第一层是浏览器，第二层是 Web 服务器，第三层是数据库服务器。这种结构的核心是 Web 服务器，它负责接收远程 (或本地) 的 HTTP (超文本传送协议) 数据请求，然后根据查询条件到数据库服务器获取相关的数据，并把结果翻译成 HTML (超文本置标语言) 文档传送给提出请求的浏览器。在三层结构中，数据库服务器完成所有的数据操作，Web 服务器则负责接收请求，然后到数据库服务器中进行数据处理，然后再对客户机给予答复。

例如，采用 WAMP 进行开发的数据库系统，Web 服务器是 Apache，数据库服务器是 MySQL。ASP.NET(C#) 是当前流行的 Web 程序设计工具，它运行于 IIS (Internet Information Server, 是 Windows 开发的 Web 服务器)，ASP.NET(C#)+MySQL 也是一个开发 Web 数据库的好组合。此外，Java 作为跨平台的面向对象的程序设计语言，最常用的 Web 服务器有 Tomcat，其他还有 JBoss、Web Logic 等，Java+MySQL 也是一个开发 Web 数据库的好搭档。

例如，广州番禺职业技术学院顶岗实习系统是为管理毕业生顶岗实习而编写的管理系统，该系统基于 B/S 模式。图 1.8 所示是系统的登录界面，用户用初始密码登录后可以完善个人信息，也可以更改登录密码。不同用户执行不同的操作权限，例如，教务处老师录入学生信息、教师信息，学生填写实习信息、实习企业信息、实习周记，教研室主任上传实习计划，安排实习老师，系主任可以查看实习信息汇总、实习公司采集以及统计报表。此外，系统还包括校内老师评价学生、校外老师评价学生和论文指导等功能模块。