

项目 1 数据库基础和认识 SQL Server

数据库是计算机软件的重要内容,它为应用程序提供数据的存取服务。

本书使用 SQL Server 数据库开发学生成绩管理系统,先要明确项目的内容以及数据库系统开发的过程,并有一定的知识准备,还要为后面的项目安装 SQL Server。

学生成绩管理系统存储和管理学生信息、课程信息和成绩信息,并保证信息的准确性;同时,系统能够对以上信息进行查询、检索;系统还要能够对这些数据进行安全管理和日常维护。

数据库系统开发是软件系统开发的一部分,并伴随着软件系统开发和应用的全过程,如图 1-1 所示。数据库设计不是本书的重点,只进行简单介绍,数据库设计完成得到数据库和表则是必需的,各阶段的具体内容在下面用箭头标出。在掌握必要的数据库基础知识,并且安装 SQL Server 之后,将基本按照数据库系统开发过程开发学生成绩管理系统。

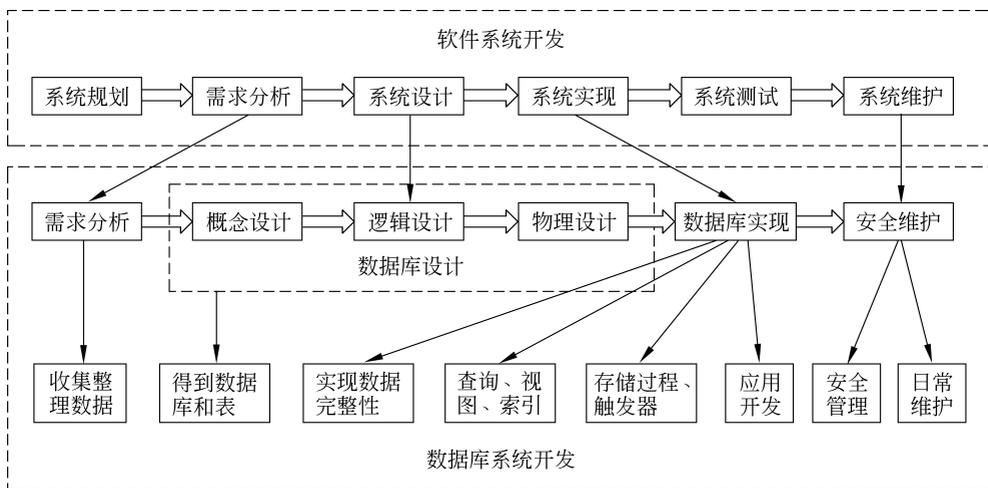


图 1-1 软件系统开发过程和数据库系统开发过程

本书主要内容就是数据库的创建、管理和应用,具体是指:创建数据库,并在其中创建表;然后管理表里面的数据(数据的增加、删除、修改和查询),为了提高效率,使用了视图、索引、存储过程等手段;当然,在管理数据的过程中用到一些工具(约束、触发器等)来保证数据的准确和合理(数据完整性)。

本项目涉及的知识点和任务如图 1-2 所示。

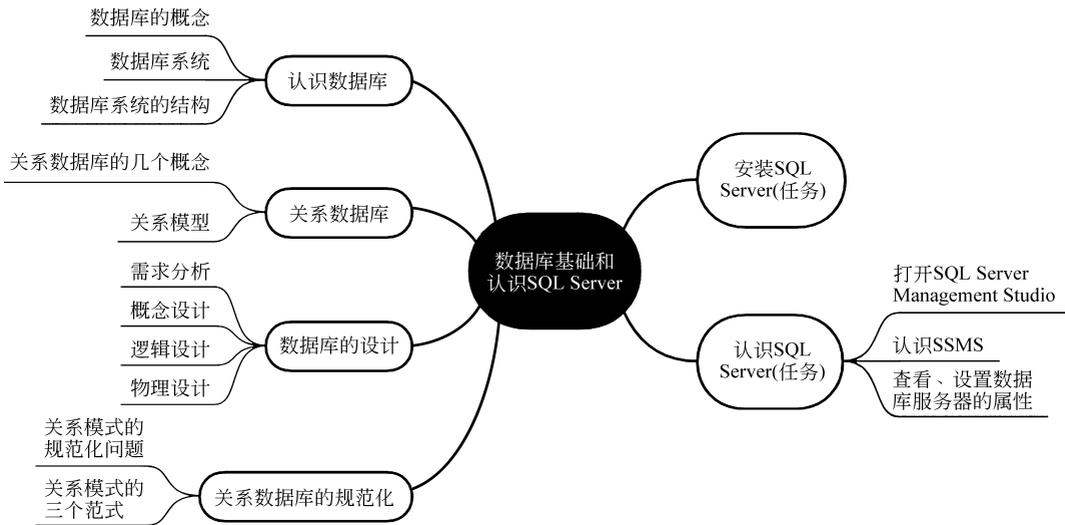


图 1-2 项目 1 思维导图



项目目标

- 理解掌握数据库的基本概念,重点是 E-R 图。
- 理解掌握关系数据库的规范化设计。
- 了解我国数据库发展的现状,增强民族自信。
- 了解 SQL Server 的安装过程。
- 学会 SQL Server 组件中常用管理工具的使用。

1.1 知识准备

知识 1-1 认识数据库

数据库技术是信息时代技术基础之一,数据库技术已经广泛应用到日常生活当中,比如网上购物,12306 网站购买火车票,超市购买商品时采用 POS 机结账,电信公司对电话费的管理,等等。当然这些例子中也应用了除数据库技术以外的其他计算机技术,比如网络技术等。



知识 1-1

1. 数据库的概念

什么是数据库(database,DB)? 简单来说就是放数据的仓库,数据库中的数据可以有数字、字母、文字等符号,还可以有图片、声音、视频等信息。仓库中的物品存放是有条理的,数据库中的数据也是有组织的,同时也是可以共享的。为了更好地理解什么是数据库,把数据库比作图书馆,并在表 1-1 中进行对比。当然这里所说的图书馆是指存放纸质媒介,不包括电子图书的图书馆。

表 1-1 图书馆与数据库的对比

对比项	图书馆	数据库
存放内容	图书、报刊	数据
存储介质	纸张、书架	计算机文件
有序性	分门别类	按照一定的方式组织起来
共享性	一本书只能被一个读者借阅	数据可以被多个用户共享
管理	只能新增图书或报废图书,不能修改	数据可以增加、修改和删除

数据库是存放有组织、可共享的数据集合,需要通过数据库管理系统(database management system,DBMS)来管理数据,比如,常用的数据库软件 SQL Server、MySQL、Oracle 等就是数据库管理系统。数据库管理系统使用结构化查询语言(structured query language,SQL)来实现数据定义功能、数据操作功能和维护数据安全的功能。当然,具体完成数据库的管理工作是离不开人的参与的,负责数据库的建立、使用和维护的专门人员就是数据库管理员(database administrator,DBA)。由数据库、数据库管理系统、数据库管理员、用户以及操作系统和相应的硬件等组成的计算机系统称为数据库系统,比如,本书要开发的学生成绩管理系统。

2. 数据库系统

(1) 数据管理技术的发展过程。数据库是一种计算机数据管理技术。为了更好地理解数据库的概念,需要从数据管理技术的发展过程讲起。在计算机引入数据处理领域后,数据管理技术的发展经历了三个阶段:人工管理阶段、文件系统阶段和数据库系统阶段。数据库之前的数据管理技术如表 1-2 及图 1-3 和图 1-4 所示。

表 1-2 数据库之前的数据管理技术

发展阶段	所处时期	特点
人工管理	20 世纪 50 年代中期以前	数据不保存;数据面向应用,不能共享; 数据由应用程序进行管理,数据与程序不具备独立性
文件管理	20 世纪 50 年代后期至 60 年代中期	数据保存在文件中,可以查询、插入、删除和修改; 数据文件由文件系统进行管理,程序与数据的独立性仍较差; 易造成数据的不一致性;数据冗余度大(数据大量重复的现象,称为冗余)

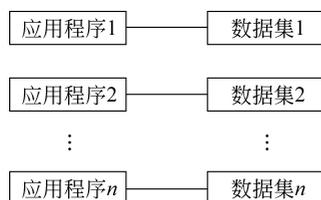


图 1-3 数据的人工管理

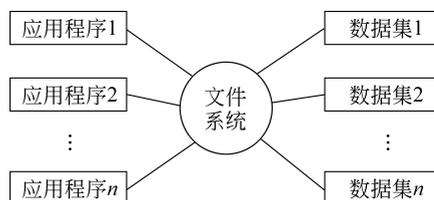
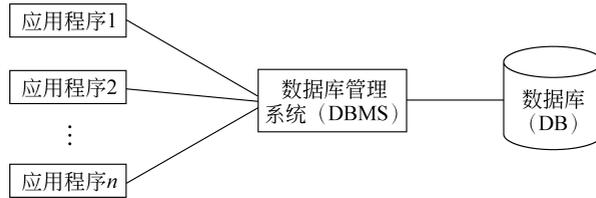


图 1-4 数据的文件管理

(2) 数据库系统的特点。数据库技术诞生于 20 世纪 60 年代末期,相比于数据库之前的数据管理技术,数据库系统不再针对某一应用,因而具有整体的结构化;同时能从整体和全局上看待和描述数据,极大地减少了数据冗余;还提高了数据的共享性和独立性;数据库管理系统也能够对数据进行统一的管理和控制,如图 1-5 所示。



注意：数据库通常在图形中用圆柱体表示。

数据库系统的特点如表 1-3 所示。

表 1-3 数据库系统的特点

特 点	说 明
数据结构化	数据的组织方式、存取方式由 DBMS 统一管理
数据冗余度小	数据共享节约了存储空间,避免了数据之间的不相容性与不一致
数据共享性好	数据可以被多个用户和多个应用共享使用
数据独立性高	数据在物理上和逻辑上都独立于应用程序
数据库保护	数据库管理系统能够对数据进行统一的管理和控制,包括数据的安全性、完整性、并发控制与故障恢复

3. 数据库系统的结构

从数据库系统应用的角度看,目前数据库系统常见的结构有客户机/服务器(client/server,C/S)结构和浏览器/服务器(browser/server,B/S)结构。

(1) C/S 结构。C/S 软件系统体系结构中客户机完成业务处理,数据表示以及用户接口功能;服务器完成 DBMS(数据库管理系统)的核心功能,如图 1-6 所示。

C/S 结构的优点是能充分发挥客户机 PC 的处理能力,减轻服务器运行数据的负荷;缺点是客户机需要安装专用的客户端软件,对客户端的操作系统一般也会有限制,系统软件升级时,维护和升级成本非常高。

在客户机和服务器中间增加一个应用服务器,分担客户机的业务处理,称为中间层,构成 3 层结构,如图 1-7 所示。

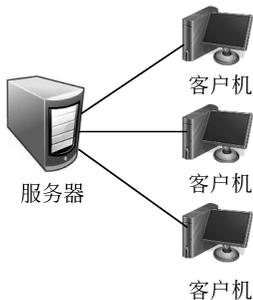


图 1-6 C/S 结构示意图

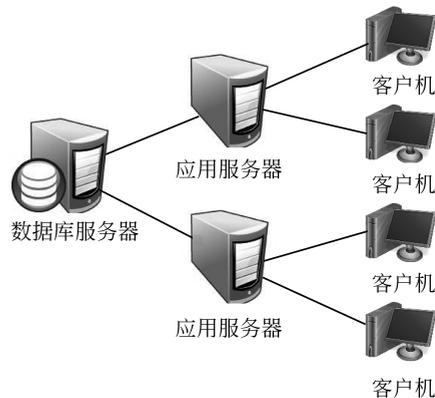


图 1-7 3层 C/S 结构示意图

(2) B/S 结构。B/S 可以看作特殊的 C/S 结构。Browser 指的是 Web 浏览器,实现极少数事务逻辑,主要事务逻辑在服务器实现。B/S 结构的优点是客户端无须安装,方便维护和业务扩展;缺点是速度和安全性不理想。B/S 结构应用广泛,典型的例子有网上订票、购物等,如图 1-8 所示。



图 1-8 B/S 结构示意图

知识 1-2 关系数据库



知识 1-2

1. 关系数据库的几个概念

(1) 实体(entity)。数据是对客观世界的抽象描述,为了描述的方便,把客观存在的互不相同的事物称为实体,比如学生、教师、课程等。

(2) 实体集(entity set)。同类型的实体集合称为实体集,比如某个学校的所有学生、所有教师等。

(3) 属性(attribute)。实体具有的某一个特性称为属性,比如学生实体的学号、姓名、性别等。

(4) 键(key)。实体既然是互不相同的,这种区别就要通过其属性的不同来体现,即使在实体集里也是如此。能够唯一地标识实体的最小的属性组称为实体的键或者称为码。例如,在学生实体集中,一个学号可以唯一地对应一个学生,学号就学生实体的码或者键。如果一个学生只有一个手机号码,并且互不重复,那么手机号码也可以作为学生实体的键或者码;如果选择学号,学号就是学生实体集的主键,主键只能有一个。手机号码就是候选键。实体的互不相同,可以理解为哲学上的“世界上没有两片完全相同的树叶”。

(5) 联系(relationship)。哲学上说世界是普遍联系的,实体和实体之间也是有联系的,比如,学生实体和课程实体是通过“选修”联系起来的,教师实体和课程实体之间是通过“授课”联系起来的。

实体之间的联系有一对一联系(1:1),一对多联系(1:n)和多对多联系(m:n)。例如,班级和班长(正班长)之间就是一对一联系,一个班级里有一个班长,一个班长对应一个班级。班级和学生之间就是一对多联系,一个班级里有多个学生。学生和课程之间是多对多联系,学生选修多门课程,课程有多个学生学习。

2. 关系模型

数据库是有组织的数据集合,如何组织,就要看数据库采用什么样的数据模型来描述实体及其联系。主要的数据模型有层次模型、网状模型和关系模型。目前,采用关系模型的数据库系统应用最为广泛,比如本书所使用的 Microsoft SQL Server,还有 MySQL、Oracle、

DB2 等。为了叙述简洁,后面所讲的数据库就是指关系数据库。

关系模型是以二维表来描述实体和实体的联系的,“关系”两个字是《离散数学》中集合论的一个数学概念,有需求的读者可以查阅相关内容,本书作为数据库应用的基础读物予以省略,敬请读者注意。

二维表之所以称为二维,是因为有行有列,其中的行描述实体,列描述实体的属性。在数据库中,二维表称为表(table),表的行称为记录(record),表的列称为字段(field),如图 1-9 所示。图中的学生表可表示学生实体及其属性。

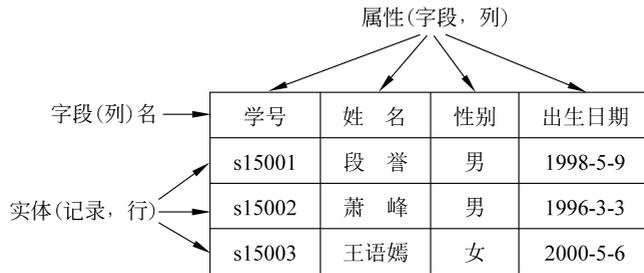


图 1-9 表的结构

实体之间的联系也是用表来表示的。实体的一对一联系直接在一张表里表示出来。例如,班级表里有班级编号、班级名称和班长学号。实体的一对多联系用两张表表示。例如,学生表和班级表。实体的多对多联系需要增加一张表来表示联系本身。例如,学生和课程之间的选修联系。选修表中除了学号和课程编号,还有选修这个联系的属性——成绩,所以,选修表就是成绩表,如图 1-10 所示。实际上,成绩表将多对多联系转化为两个一对多联系了(1 个学生对应多门课的成绩,1 门课程对应多个学生的成绩)。

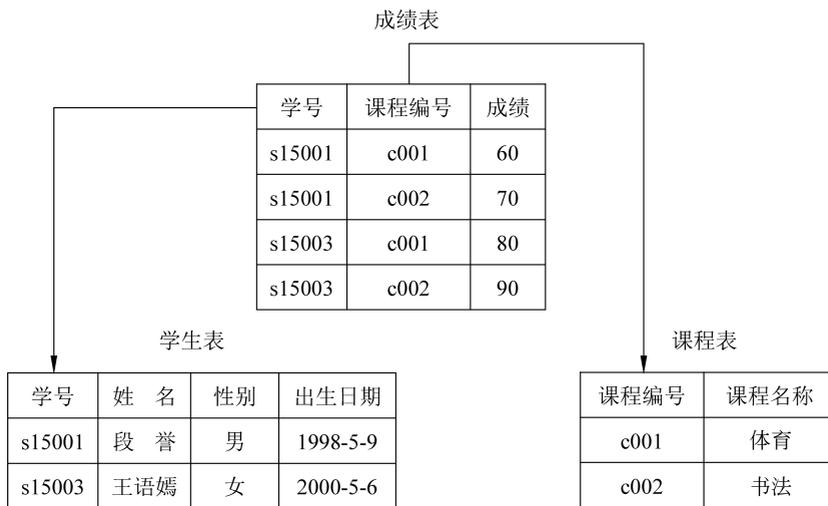


图 1-10 用表来表示实体之间的联系

知识 1-3 数据库的设计



知识 1-3

在图 1-2 数据库系统开发过程中有数据库的设计步骤。需求分析是软件系统开发的重要阶段,本节强调针对数据库设计的需求分析内容,所以将需求分析放在数据库设计的准备阶段进行介绍。

1. 需求分析

需求分析是分析系统的需求,主要任务是调查、收集与分析用户在数据管理中的信息需求、处理需求和安全性与完整性的需求,并把这些需求写成需求说明书。

(1) 本书的“学生成绩管理系统”,对学校教务处的成绩管理流程进行梳理,主要功能如下。

- ① 学生管理能够查询、修改、添加和删除学生的基本信息。
- ② 课程管理能够修改、添加新开课程,删除淘汰课程。
- ③ 成绩管理要记录学生每门课的成绩并提供查询、修改和简单的统计功能。

(2) 系统中有两个实体,即一是学生,二是课程。

(3) 每个实体对应的属性如下。

- ① 学生实体的属性:学号,姓名,性别,出生日期。
- ② 课程实体的属性:课程编号,课程名称。

(4) 该系统的规则是:一个学生可以选修多门课程,一门课程也可以被多个学生选修。

2. 概念设计

概念设计是将得到的用户需求(要描述的现实世界)抽象为概念数据模型,其过程是首先根据单个应用的需求,画出能反映每一个需求的局部实体—联系模型(entity-relationship, E-R 图)。然后把这些 E-R 图合并起来,消除冗余和可能存在的矛盾,得出系统全局的 E-R 模型。需要特别强调的是,画 E-R 图的基础就是需求说明书中的数据流图和数据字典。

E-R 图中矩形表示实体;椭圆形表示属性;菱形表示联系;然后用短线连接在一起。如图 1-11 所示是学生实体、课程实体、选修联系的局部 E-R 图,将局部 E-R 图连接起来成为图 1-12 所示的全局 E-R 图,图中标注了学生和课程之间是多对多联系($m:n$)。其中加下划线的属性是实体或者联系的键,如学生实体的学号,课程实体的课程编号。

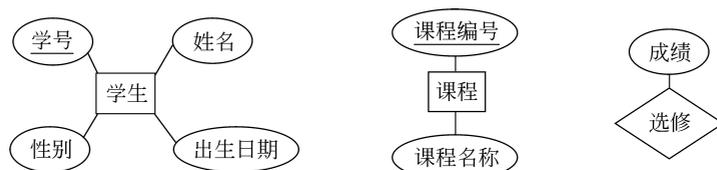


图 1-11 学生实体、课程实体、选修联系的 E-R 图

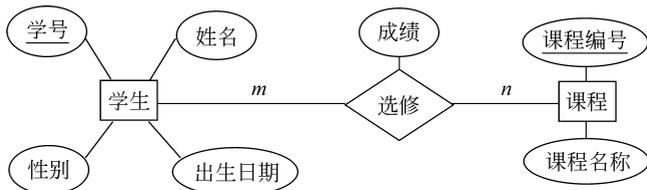


图 1-12 全局 E-R 图

3. 逻辑设计

概念设计是从设计者的角度来分析问题,是让设计者能够清楚认识系统结构的理解过程,要实现系统,需要把概念模型转换为具体计算机上 DBMS 所支持的结构数据模型。

逻辑设计就是把 E-R 图转化为关系模式,即把实体及其联系转化为关系模式,然后对关系模式进行优化。图 1-13 是 E-R 图转化为关系模式后在 DBMS 上实现的示意图。图中左边表示 E-R 图,按箭头方向转化为关系模式,就是中间的二维表,然后在数据库管理系统(DBMS)的帮助下进入数据库,从而实现逻辑设计。

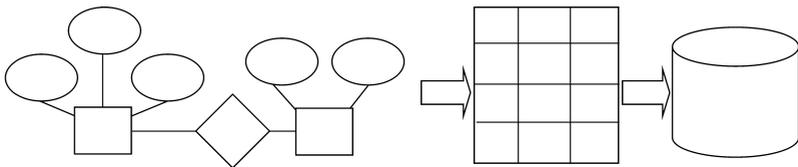


图 1-13 E-R 图转化为关系模式后在 DBMS 上实现

E-R 图转化为关系模式要遵循一定的规则,转化产生的关系模式可能是不合理的,比如,存在数据冗余和插入、删除、更新异常,解决的方法是关系模式的规范化。

4. 物理设计

物理设计主要确定数据库的存储结构,包括确定数据库文件和索引文件的记录格式和物理结构,以及选择存取方法等,基本上由数据库管理系统完成。

物理设计阶段的主要内容是创建和管理数据库和表,学生成绩管理系统数据库的物理设计将在后面的项目中逐步进行。

知识 1-4 关系数据库的规范化

1. 关系模式的规范化问题

假设初学者将图 1-12 转换为如下关系模式:学生(学号,姓名,性别,出生日期,课程编号,课程名称,成绩),会产生什么问题呢?

(1) 数据冗余: 在上面的关系模式中,如果一个学生选修了 n 门课,那么这个学生的姓名会在这个关系模式中出现 n 次;同样一门课如果被 m 个学生选修,那么这门课的名称也会在这个关系模式中出现 m 次。



知识 1-4

(2) 插入异常：如果新增加一门课，还没有学生选修，上面的关系模式中会在属性学号、姓名、性别、出生日期上没有值，在数据库中就是“空值”，这样会引起检索和操作的不便。更为严重的是，属性学号是上面关系模式的键的一部分，是不允许出现“空值”的。

(3) 删除异常：如果某个学生中途退学，从上面的关系模式中删除该学生的信息，结果会将该学生选修过的课程信息一同删除。这种情况是不合适的，因为可能会丢失课程信息。

(4) 更新异常：如果发现某门课程的课程编号弄错了，需要修改，那么这门课所有弄错的课程编号都要修改。如果不慎漏掉了几个，就会引起数据的不一致。

2. 关系模式的三个范式

上面的关系模式很明显是不好的，好的关系模式就是消除存在数据冗余和插入、删除、更新异常。如何消除呢？关系模型的奠基人 E. F. Codd 系统地提出了第一范式 (first normal form, 1NF)、第二范式 (2NF) 和第三范式 (3NF)，来讨论怎么将不好的关系模式转化为好的关系模式，这就是关系模式的规范化。

所谓“第几范式”，是表示关系模式的某一级别，有高低之分，1NF 最低，2NF 次之，后面以此类推。范式之间是包含关系，满足 2NF 关系模式必然满足 1NF，满足 3NF 关系模式必然满足 2NF，当然也满足 1NF，如图 1-14 所示。3NF 后面还有 4NF、5NF，一般情况下，关系数据库只需满足 3NF 就可以了。

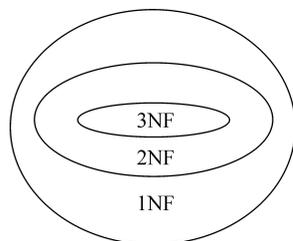


图 1-14 关系模式三个范式的包含关系

(1) 1NF。当关系模式的每个属性值都是不可再分的最小数据单元时，则满足 1NF，这显然是关系模式的基本要求。如果上面的学生关系模式中的姓名属性出现两个值，如“萧峰”和“乔峰”，那么姓名属性就是可以分解的，可以分解为姓名和曾用名，这样才能满足 1NF。

(2) 2NF。当关系模式满足 1NF，并且非主属性完全依赖于主键（不存在部分依赖）时，就满足 2NF。比如，上面的学生关系模式中的姓名只依赖于主键（学号，课程编号）中的一部分（学号），而不是整个主键，所以不满足 2NF。规范化的方法是：将部分依赖的属性和被部分依赖的主属性从原关系模式中分离，形成一个新的关系模式。具体是将上面的学生关系模式分解成下面的学生、课程和成绩三个关系模式就都满足 2NF，当然分解的结果也满足 3NF。

学生(学号, 姓名, 性别, 出生日期)

课程(课程编号, 课程名称)

成绩(学号, 课程编号, 成绩)

E-R 图中的选修联系转化为关系模式成绩，更符合习惯。成绩既与学生有关，也与课程有关，所以增加代表两者的学号和课程编号。学号或者课程编号不能单独作为主键，也就是说成绩表中的学号或者课程编号是有重复的，但是两者合在一起不会重复，所以它的主键是这两个属性的组合。从整体来看，学号和课程编号是冗余的，但这是必需的，比起不满足 1NF 的学生(学号, 姓名, 性别, 出生日期, 课程编号, 课程名称, 成绩)关系模式，冗余小得多。冗余不能完全消除，否则影响数据库的正常运行。

(3) 3NF。当关系模式满足 2NF，并且非主属性仅依赖于主键（不存在传递依赖）时，就满足 3NF。例如，学生(学号, 姓名, 性别, 出生日期, 所在系, 系主任)是符合 2NF 的，但存在

传递依赖,所在系依赖于学号,而系主任又依赖于所在系,所以系主任传递依赖于学号,因此不满足 3NF。规范化的方法是:将具有传递依赖的属性从原关系模式中分离,形成一个新的关系模式。具体是将例子中的学生关系模式分解成学生(学号,姓名,性别,出生日期,所在系)和系(系名称,系主任)两个关系模式,就都满足 3NF 了。

因为关系模式的规范化决定了数据库能否正常运行,所以必须认真对待。从上面规范化的过程来看,首先,要将实体的属性细分为不可再分的状态;其次,要将不同的实体和联系拆分成不同的关系模式;最后,适当调整关系模式中的属性,避免属性之间出现部分依赖和传递依赖。这样就能满足规范化的要求了。

1.2 任务划分

任务 1-1 安装 SQL Server

提出任务

先检查安装前的软硬件环境是否满足要求,然后按步骤安装。

实施任务

本书安装的是 Microsoft SQL Server 2012 Enterprise Edition Service Pack 1,操作系统是 Windows 7。Microsoft SQL Server 有很多版本,高版本相比于低版本性能更好,功能也更丰富,但是基本应用大同小异,所以本书没有选择最新版本,而是选择具有一定市场的 2012 版,作为基本的应用完全够用。

安装的软件环境推荐: Windows 7、Windows Server 2008 Service Pack 2 及以上版本的 32 位或者 64 位操作系统;硬件环境推荐: 主流 CPU,内存 2GB 及以上,硬盘空间 6GB 及以上。

(1) 从运行安装包中的 setup.exe 开始,打开“SQL Server 安装中心”窗口,如图 1-15 所示。

(2) 单击图 1-15 所示窗口左侧的“安装”,进入安装界面,选择安装类别,如图 1-16 所示。

(3) 单击图 1-16 所示的“全新 SQL Server 独立安装或向现有安装添加功能”,接下来按向导安装程序支持规则,输入密钥,接受许可条款,安装程序支持文件,然后进入“功能选择”界面,选择要安装的组件,如图 1-17 所示。

功能选择步骤里面,如果不需要分析服务 Analysis Services、报表服务 Reporting Services、集成服务 Integration Services,以及 SQL Server Data Tools,可以不选择,像“SQL Server 复制”、客户端的工具××、管理工具××等必须选择。其中“管理工具—基本”“管理工具—完整”对应的是 SQL Server 2012 的 SQL Server Management Studio。

(4) 下一步是“实例配置”界面,如图 1-18 所示,选择“默认实例”选项。简单地说,实例是实际的数据库例子,其实就是 SQL Server 数据库引擎。同一台计算机上可以同时运行多个实例,并且互相独立。