

单元一 分析与设计数据库



学习目标

- 创建实体关系模型
- 分析实体的关系类型
- 设计数据库
- 规范化数据

任务 1.1 分析数据库

1.1.1 情景描述

某高校要开发一套学生信息管理系统，以实现教学管理的信息化、规范化、科学化。该系统实现的主要功能包括：使用计算机对教学活动中的各种信息进行记录和管理，记录专业、系部、班级、学生、教师、课程、学期、辅导员评价等基本信息，比如学生信息需要记录学生的学号、姓名、性别、生日等，课程信息记录课程的课程号、课程名、学分、学时等；同时能够对这些信息进行增加、删除、修改、查询等操作；能够按照日期、上课节次记录学生的迟到、早退、事假、病假情况，最终生成每周考勤记录表；能够对学生的选课信息进行管理，记录学生所选择的课程、课程由哪些老师教授、学生所选课程的成绩等；能够让辅导员在每个学期期末对学生进行评价……那么应该如何实现这个系统，使它在对学生、教师等使用者提供业务功能的同时，还要对数量庞大、关系复杂的各类数据进行管理？

1.1.2 问题分析

早期的软件开发主要用于科学计算，在程序运行时输入数据，运算处理结束后得到结果数据输出。随着计算任务的完成，数据和程序会一起从内存中释放，没有特别的保存数据，也就不需要使用数据库来管理程序中用到的数据。如今的计算机存储的数据量非常庞大，数据在多个程序之间可以共享，并且还会频繁地对存储的数据进行操作，这就需要使用数据库对这些数据进行统一管理。因此面对这个系统，作为数据库分析人员，首先应对学生信息管理系统需要处理的数据进行收集，分析每个数据的特征、数据之间存在的关系以及定义的规则，收集以往纸质的、手工的学生信息管理档案，根据收集到的数据，确定实体、实体的属性及实体之间的关系，画出实体关系图，通过该图能很好地进行开发人员和用户之间的沟通交流，并指导后续数据库方面的工作。

1.1.3 解决方案

1. 识别实体

在学生信息管理系统中，经过与客户交流分析，得出主要有以下几类实体：存储专业信

息的专业实体、存储班级信息的班级实体、存储学生信息的学生实体、存储辅导员评语的辅导员评语实体、存储课程信息的课程实体、存储教师信息的教师实体、存储系部信息的系部实体。

2. 识别实体的属性

(1) 专业实体的属性有专业代码、专业名称、专业的状态信息等,如图 1-1 所示。

(2) 班级实体的属性有班级代码、班级名称、所属专业、所属年级、班主任等,如图 1-2 所示。

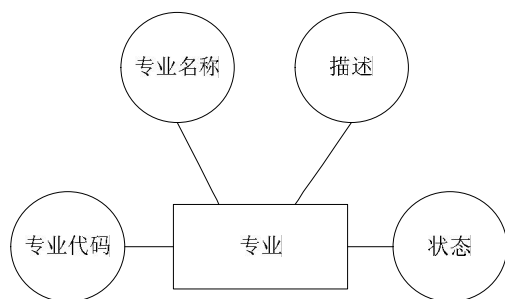


图 1-1 专业实体的属性

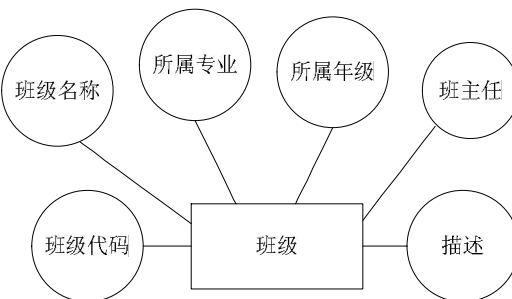


图 1-2 班级实体的属性

(3) 学生实体的属性有学号、姓名、性别、出生年月等,如图 1-3 所示。

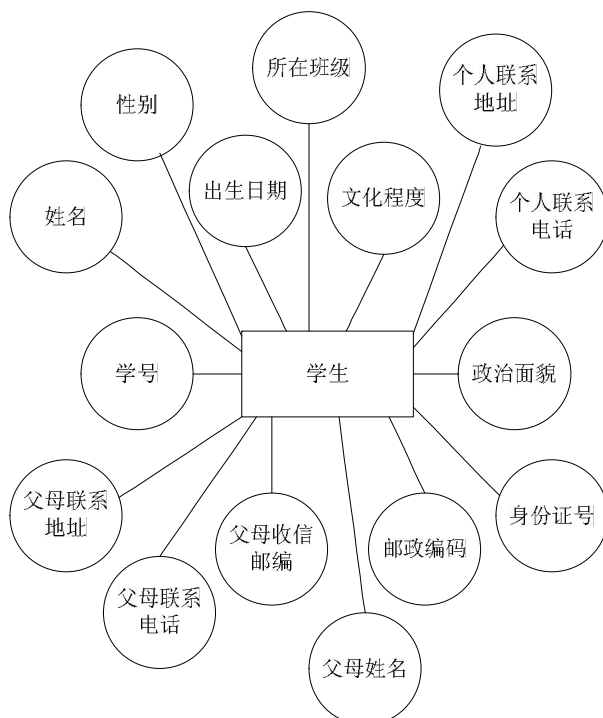


图 1-3 学生实体的属性

(4) 辅导员评语实体的属性有评价代码、学号、学期代码、评价寄语等,如图 1-4 所示。

(5) 课程实体的属性有课程编号、课程名称、课程性质、学分、开课学期、课程分类,如图 1-5 所示。

(6) 教师实体的属性有教师编号、教师姓名、性别、职称、专业、学历,如图 1-6 所示。

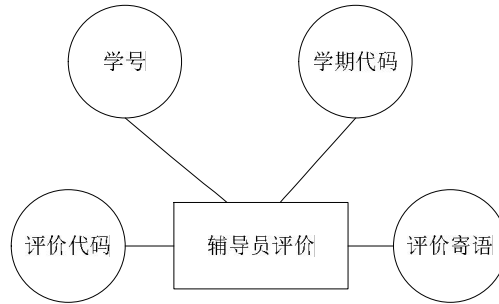


图 1-4 辅导员评价实体的属性

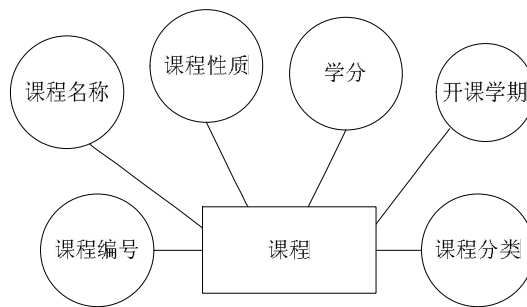


图 1-5 课程实体的属性

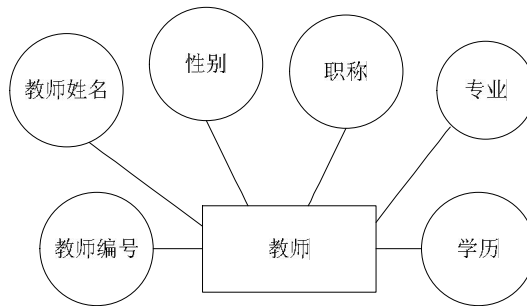


图 1-6 教师实体的属性

(7) 系部实体的属性有系部代码、系部名称、办公电话、系主任等，如图 1-7 所示。

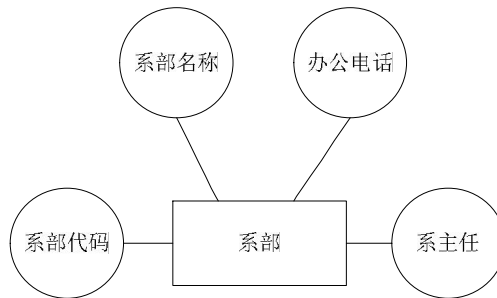


图 1-7 系部实体的属性

3. 识别实体之间的联系

一个班级只能属于一个专业，而一个专业可以包含很多个班级，所以班级实体和专业实

体之间是多对一的关系。

一个学生只能在一个班级，一个班级可以包含很多个学生，所以学生实体和班级实体之间是一对多的关系。

一个学生可以选修多门课程，一门课程可供多个学生选修，所以学生实体和课程实体之间是多对多的关系。

一个辅导员评价只能适应于一个学生，而一个学生可以由多个辅导员评价，所以学生实体和辅导员评价实体之间是一对多的关系。

一个教师可以对多门课程进行授课，一门课程也可以由多个老师来任教，所以课程实体和教师实体之间是多对多的关系。

一个教师可以对多个班级进行授课，一个班级也可以由多个老师来任教，所以教师实体和班级实体之间是多对多的关系。

一个系部有多个老师，一个老师只能属于一个系部，所以系部实体和老师实体之间是一对多的关系。

根据以上描述，该系统实体之间的关系如图 1-8 中的 E-R 模型所示。

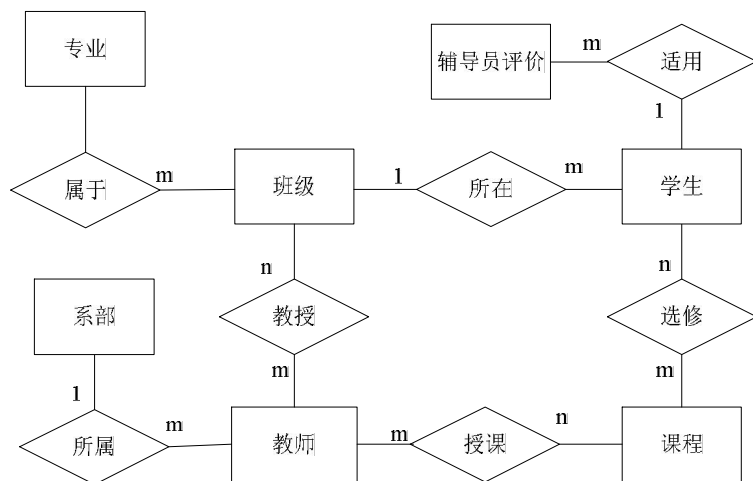


图 1-8 系统 E-R 模型

1.1.4 知识总结

1. 数据库基本知识

数据有各种表现形式，如数字、图像、文字、声音等。数据必须经过数字化过程，才能被计算机识别、存储、处理。那么简单地讲，数据库（DataBase, DB）就是存放数据的仓库，是将软件开发中涉及到的数据按照一定的组织形式存储在一起的相互关联的数据集合。数据库中存储的基本对象就是数据。

数据库管理（DataBase Management）是对数据库进行维护以便信息能够正常使用。

数据库管理系统（DataBase Management System, DBMS）是对数据进行管理的大型系统软件，是提供存储数据和检索数据的软件，是数据库系统的核心部分。

关系型数据库管理系统（Relational Database Management System）在数据库管理系统（DBMS）的基础上增加关系，通过对数据、数据的关系及数据的约束来存储和管理数据。

2. 概念模型

对于某一应用环境所涉及的数据以及数据的联系进行抽象，形象地用一种模型的方式表达我们要保存的数据以及数据之间的关系，以供数据库设计人员和用户之间进行沟通交流，这样的模型称为概念模型。该模型按照用户的观点来对现实世界建模，完全不涉及信息在计算机中的表示，只是用来描述某个特定对象所关心的信息结构。目前，有很多流行的概念模型表达方式，最常用的方法是 Peter Chen 于 1976 年提出的实体—联系方法（简称 E-R 方法）来描述数据库中需要存储的数据及它们之间的关系，它按用户的观点、通过图示法来描述信息结构。应用 E-R 方法建立的概念模型称为 E-R 模型，E-R 图则是直接表示概念模型的工具。

（1）实体。

实体是数据模型中的一个概念，我们把客观存在并且相互区别的事物称为实体。实体可以是具体的一个学生、一件商品、一张桌子等，也可以是抽象的事件，如学生选修课程、教师教授课程等。在软件开发中需要保存的所有对象都可以称为实体，在 E-R 模型中用矩形框表示，矩形框内写上实体名。如超市销售业务系统中的商品信息、供应商信息、顾客信息等都可以称为实体，如图 1-9 所示。

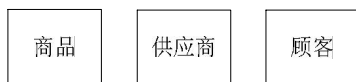


图 1-9 实体

（2）属性。

能够描述实体的特征的就是实体的属性，一个实体可以由多个属性共同描述。在 E-R 模型中用椭圆框表示，在框内写上属性名，并且用直线与对应的实体相连。如确定一个商品实体，需要知道商品的编号、名称、单价、库存数量、生产日期、保质期等，这些都可以称为商品实体的特征或者属性。供应商实体的特征有供应商编号、供应商名称、联系方式、地点、信誉等。顾客实体的特征有顾客编号、顾客姓名、性别、年龄、职业、联系方式、地址、办卡时间、积分等，如图 1-10 所示。

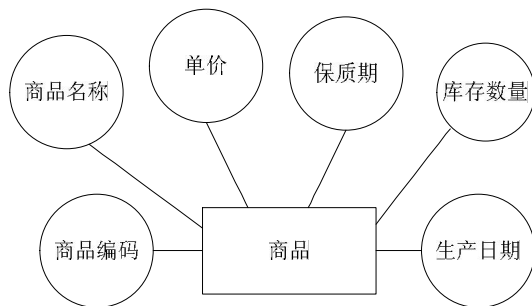


图 1-10 属性

（3）关系。

每个实体不是孤立存在的，它们之间有这样或者那样的关系，在 E-R 模型中用菱形表示，菱形框内标注关系的名字，并用直线与有关实体相连。如学生和课程之间有关系，记录一个学生选的是哪几门课程、这个课程被哪些学生选，这个关系可以命名为“选课”关系。

根据实体之间的对应关系，可以把关系分为以下三种类型：

- 一对一 (1:1): 两个实体集合中的数据之间是一一对应的关系。对于实体集 A 中的每个实体, 实体集 B 中最多有一个与之相对应; 反之, 对于实体集 B 中的每个实体, 实体集 A 中最多也只有一个与之相对应。如系部实体和系主任实体, 一个系部只能有一个系主任, 一个系主任只能领导一个系; 身份证实体和公民实体之间的关系也是一一对应的, 如图 1-11 所示。
- 一对多 (1:m): 两个实体集合中的数据之间是一对多的关系。对于实体集 A 中的每个实体, 实体集 B 中有多个与之相对应; 反之, 对于实体集 B 中的每个实体, 实体集 A 中最多有一个与之相对应。如班级实体和学生实体, 一个班级可以有多个学生, 但一个学生只能分在一个班级, 那么班级实体和学生实体之间是一对多的关系; 系部实体和教师实体之间的关系也是一对多的关系, 如图 1-12 所示。

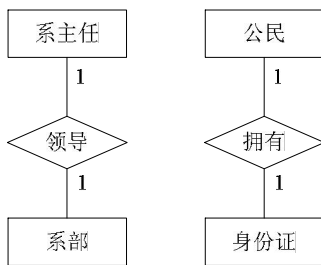


图 1-11 一对一

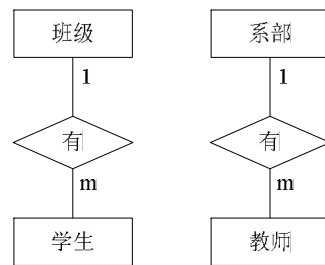


图 1-12 一对多

- 多对多 (m:n): 两个实体集合中的数据之间是多对多的关系。对于实体集 A 中的每个实体, 实体集 B 中有多个与之相对应; 反之, 对于实体集 B 中的每个实体, 实体集 A 中也有多个与之相对应。如学生实体和课程实体, 一个学生可以选修多门课程, 一门课程也可以由多个学生选修, 那么学生实体和课程实体之间的关系是多对多的关系; 一个书店可以销售多种书籍, 一种书籍也可以被多个书店销售, 书店实体和书籍实体之间是多对多的关系, 如图 1-13 所示。

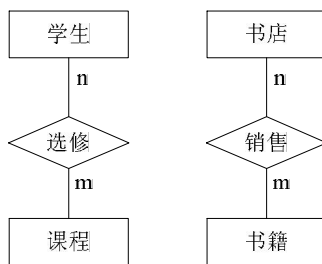


图 1-13 多对多

1.1.5 应用实践

某超市要设计一款销售业务系统, 以明确每款商品的销售情况, 来改善超市的盈利。

本系统的主要功能有用户注册和登录、商品管理、供应商管理、积分卡信息管理 4 个模块。用户注册和登录主要用于使用该系统的用户信息的注册及完成合法用户的验证、登录; 商品管理模块主要用于商品信息的添加、修改、删除、按照商品类型进行查询或者按照商品所属

的供应商进行查询；供应商管理模块主要实现供应商信息的添加、修改、删除、查询；积分卡信息管理模块主要记录顾客的积分情况，并规定一个顾客只能办一张积分卡。要求设计该超市销售业务系统的概念模型，并绘出 E-R 图。

1. 识别实体

根据上述分析，用户注册、登录功能需要有一个实体来保存用户的账号信息，那么需要一个“账号”实体。

供应商管理模块需要有一个实体来保存供应商的信息，需要一个“供应商”实体。

商品管理模块需要按照商品类型和所属供应商进行查询，那么除了已经存在的“供应商”实体外，还需要一个关于商品本身信息的“商品”实体以及一个“商品类型”实体。

积分卡信息管理模块主要记录顾客的积分情况，一个顾客只能办一张积分卡，所以可用一个“顾客”实体既保存客户信息，又保存客户的积分情况。

2. 识别实体属性

通过对需求的分析，并与用户交流，识别实体的属性如下：

(1) “账号”实体需要的属性：用户 ID、密码、邮件、用户名、地址、邮编、电话、状态。

(2) “供应商”实体需要的属性：供应商 ID、名称、地址、邮编、电话、信用状态。

(3) “商品类型”实体需要的属性：类别 ID、类别名称、描述。

(4) “商品”实体需要的属性：商品 ID、名称、价格、产品描述、生产日期、保质期。

(5) “顾客”实体需要的属性：顾客 ID、姓名、性别、年龄、职业、联系方式、地址、办卡时间、积分。顾客 ID 即是顾客办卡的积分卡编号。

3. 识别实体间的关系

(1) “供应商”和“商品”之间是多对多的关系，一个供应商可以供应多种商品，一种商品也可以被多个供应商供应。

(2) “商品”和“商品类型”之间是多对一的关系，某个商品类型下有多个商品。

(3) “顾客”和“商品”之间是多对多的关系，一种商品可以销售给多个顾客，一个顾客也可以购买多种商品。

超市销售业务系统的 E-R 图如图 1-14 所示。

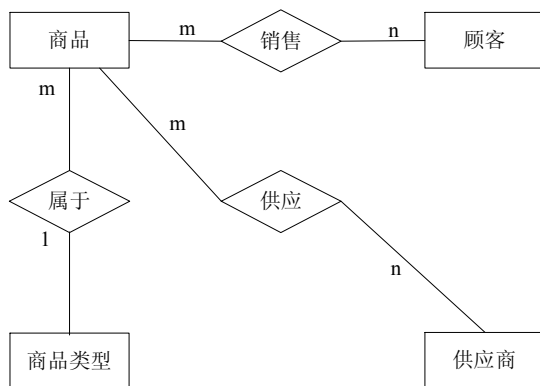


图 1-14 超市销售业务系统 E-R 图

任务 1.2 设计数据库

1.2.1 情景描述

根据分析,我们现在有了学生信息管理的概念模型(即 E-R 图),提取了在软件开发中需要保存的数据及数据的关系。这些信息需要在计算机中存储,才能支撑学生信息管理的运行。那么应该如何将这些数据和关系保存在计算机中呢?

1.2.2 问题分析

数据在计算机中的存储结构有很多种,如树型结构、图型结构、关系结构等,我们采用关系结构来存储数据,把前面得到的 E-R 图转换为表结构。

1.2.3 解决方案

1. 将实体转换为表

每个实体对应一个表,如学生实体映射成为学生表,课程实体映射成为课程表。根据 E-R 图里的实体,需要转换的实体有:专业、学期、班级、学生、辅导员评语、课程、教师。

2. 将实体的属性转换为对应表的字段

实体的每个属性都对应于表中的一列,也叫一个字段,比如学生实体中的属性学号、姓名、性别分别映射成为学生表的一列。那么本系统实体及属性对应的关系模式如下:

专业(专业代码(pk),专业名称,描述,状态)

班级(班级代码(pk),班级名称,所属专业,所属年级,班主任,描述)

学生(学号(pk),姓名,性别,出生日期,个人联系电话,政治面貌,身份证号,邮政编码,家庭联系电话,家庭联系地址)

辅导员评语(评价代码(pk),学号,学期,评价寄语)

课程(课程编号(pk),课程名称,课程性质,学分,开课学期,课程分类)

教师(教师编号(pk),教师姓名,性别,职称,学历,专业)

系部(系部代码(pk),系部名称,办公电话、系主任)

3. 将实体的联系转换为表结构

E-R 图中的关系是通过设置外键参照关系体现的,本系统具体如下:

班级实体和专业实体之间是多对一的关系,把专业实体的主键添加到班级实体的表中,作为班级实体的外键。

学生实体和班级实体之间是一对多的关系,把班级实体的主键添加到学生实体的表中,作为学生实体的外键。

学生实体和辅导员评价实体之间的关系是一对多的关系,把学生实体的主键添加到辅导员评价表中,作为辅导员评价表的外键。

学生实体和课程实体之间是多对多的关系,抽取学生实体和课程实体的主键,形成一个新的关系——选课表。

教师实体和班级实体之间是多对多的关系,教师实体和课程实体之间也是多对多的关系,分别抽取三个实体的主键形成一个新的关系——授课表。更新后的关系模式如下:

专业（专业代码（pk），专业名称，描述，状态）

班级（班级代码（pk），班级名称，专业代码（fk），所属年级，班主任，描述）

学生（学号（pk），姓名，性别，出生日期，班级代码（fk），个人联系电话，政治面貌，身份证号，邮政编码，家庭联系电话，家庭联系地址）

辅导员评语（评价代码（pk），学号（fk），学期，评价寄语）

课程（课程编号（pk），课程名称，课程性质，学分，开课学期，课程分类）

教师（教师编号（pk），教师姓名，性别，职称，学历，学位，专业，系部代码（fk））

选课（学号（fk），课程编号（fk），成绩），主键为学号和课程编号的组合。

授课（教师编号（fk），班级代码（fk），课程编号（fk）），主键为教师编号、班级代码和课程编号的组合。

系部（系部代码（pk），系部名称，办公电话、系主任）

1.2.4 知识总结

1. 数据模型

数据结构指的是数据的组成以及数据之间的联系。按照数据结构类型的不同，将数据模型划分为层次模型、网状模型和关系模型。

层次模型按照树型结构组织数据，由结点和连线组成，结点表示实体，连线表示实体之间的关系，如图 1-15 所示。这种类型的存储结构层次分明、结构清晰，有且仅有一个根结点，其他结点有且仅有一个父结点。这种模型表示一对多的关系很简便，但是表达多对多的关系比较困难。

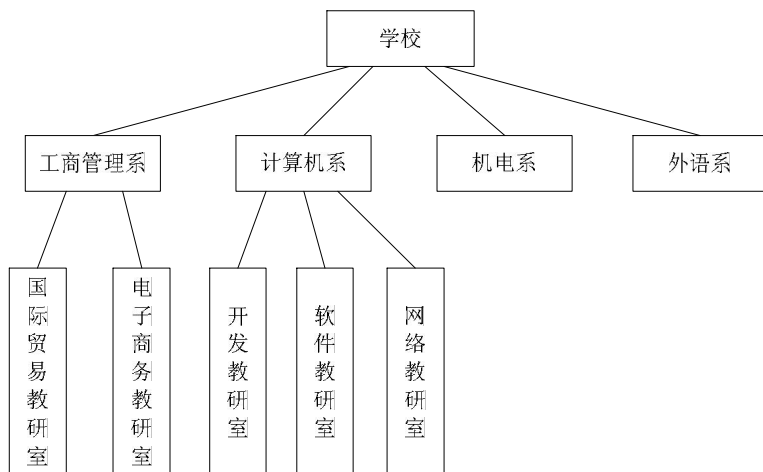


图 1-15 单位组织结构图

网状模型用网状结构表示实体以及实体之间的关系，网中结点之间的联系可以是任意的，允许多个结点没有父结点，也允许结点有多个父结点，如图 1-16 所示。这种存储结构能够很好地描述现实世界，表达结点之间的联系很方便，但是结构太复杂，不容易实现。

关系模型由 IBM 公司的 E.F.Codd 于 1970 年首次提出，在关系模型里，实体和联系均用二维表来表达，信息存放在二维表结构的表（Table）中。20 世纪 80 年代以来，计算机厂商推出的数据库管理系统几乎都支持关系型数据库。关系型数据库是基于关系模型的一种数据库，

是一些相关的表和其他数据库对象的集合。

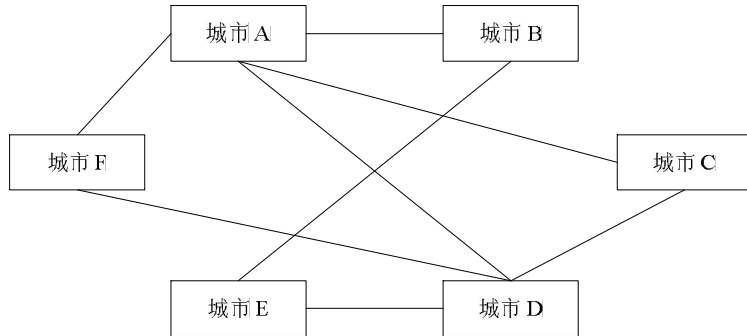


图 1-16 城市间道路

关系模型是建立在严格的数学概念的基础上的，无论是实体还是实体的关系，都用以行和列为格式的二维表格来表示，其中和行对应的是记录，和列对应的是字段。每个数据值在表中按行列排列，结构简单、灵活清晰。

关系模型可表示为：关系模型名（属性名 1，属性名 2，…，属性名 n）的形式。例如学生（学号，姓名，性别，出生日期，所在班级，文化程度，个人联系地址，个人联系电话，政治面貌，身份证号，邮政编码，父母姓名，父母收信邮编，父母联系电话，父母联系地址）。

2. 关系模型的相关概念

（1）关系（Relation）。

数据库中的每个表都有能够唯一标识其内容的表名，我们以后就把表称为关系。二维表的表名就是关系名。如表 1-1 中的表名“学生信息表”。

表 1-1 学生信息表

学号	姓名	性别	生日	民族
12180001	张英	女	1997-03-02	汉
12180002	赵刚	男	1999-02-01	汉
12180003	刘星	男	1998-06-05	回
12180004	王娜	女	1998-10-12	汉

（2）记录（Record）。

表中的每一行数据称为一条记录，又称为行或者元组。一个表中的每条记录都不能完全相同，记录在表中的排列顺序可以是无序的。如表 1-1 中的一行记录（12180001，张英，女，1997-03-02，汉）。

（3）属性（Attribute）。

二维表中的每一列称为关系的一个属性，又称为列，同一个表中的属性名不能重复，即每个列都必须有唯一的名称。如表 1-1 中的“学号”字段、“姓名”字段。

（4）域（Domain）。

表中每一列的取值范围称为此列的域，如表 1-1 中性别属性的域为（“男”，“女”）。

（5）主键（Primary Key）。

表中的每条记录都不能重复，能够唯一地标识每条记录的列或列的组合称为表的主键。

主键不能为“空”(NULL, 不是 0, 也不是空格或空字符串, 表示值不确定), 也不能重复。在学生表中, 每个学生的学号都不能重复, 我们通过学号来区分每个学生, 学生表的学号字段就称为学生表的主键。

(6) 外键 (Foreign Key)。

在选课表里, 我们要存储学生选择的课程是什么, 选课表的学号的取值范围要参照学生表的学号的取值范围, 学生表的学号字段是学生表的主键, 选课表的学号字段就称为选课表的外键, 外键可以标识一个表(选课表)和另外一个表(学生表)的关系。简单地说, 一个表(选课表)的外键学号就是另外一个表(学生表)的主键学号字段。外键的取值范围不能超出它所引用的主键的取值范围。

3. E-R 图到表的映射

(1) 实体。

能单独存在的每个实体都映射成为一张表。如学生实体用一张表来存储, 课程实体用另外一张表来存储。

(2) 属性。

E-R 图中每个实体的属性都映射成对应实体转换的表的列。如学生实体的学号、姓名、性别、出生年月分别映射成为学生表中的列。

(3) 关系。

关系映射成为表的方式依赖于关系的类型, 不同的关系映射成表的方式不同。

1) 一对一关系。

在一对一的关系中, 实体集中的每一个元素只能是一一对应的关系, 可以把两个实体对应的表合并成为一个表, 也可以将任何一个实体的主键作为另外一个实体的外键存在, 来保存他们的一一对应关系。

图 1-11 的公民与身份证的一对一的关系可以转换为如下的关系模式:

公民表(公民编号(pk), 姓名, 性别, 生日, 身份证号(fk))

身份证表(身份证号(pk), 有效开始时间, 有效截止时间, 详细地址)

系主任与系部的一对一的关系转换为:

系主任(工号(pk), 姓名, 联系方式)

系部(编号(pk), 名称, 电话, 工号(fk))

2) 一对多关系。

在一对多的关系中, 实体集合中的一个元素可以与另外一个实体中的多个元素有关系, 可以把“一”端实体的主键作为“多”端实体的外键存在, 来保存它们的一对多的关系。

在图 1-12 中, 班级与学生之间的一对多的关系可以将班级表的主键放在学生表中作为外键:

班级(班级编号(pk), 名称, 年级)

学生(学号(pk), 姓名, 性别, 生日, 班级编号(fk))

系部与教师之间的一对多的关系可以将系部的主键放在教师表中作为外键:

系部(编号(pk), 名称, 电话)

教师(工号(pk), 姓名, 职称, 电话, 系部编号(fk))

3) 多对多关系。

在多对多的关系中, 两个实体的元素是多对多的关系, 可以把两个实体的主键和多对多

关系自身的特征组合成为一张新表，新表的主键就是两个表的主键的组合。

图 1-13 中，学生与课程之间的多对多的关系可以转换为一个新的关系模式——选课关系，如下：

学生（学号（pk），姓名，性别，生日）

课程（课程编号（pk），课程名称，课程性质，学分，开课学期，课程分类）

选课（学号（fk），课程编号（fk），成绩）

书店和书籍之间的多对多的关系转换为销售关系，如下：

书店（编号（pk），名称，地址，联系方式）

书籍（ISBN（pk），书名，作者，定价，出版社，出版日期）

销售（编号（fk），ISBN（fk），销售数量）

1.2.5 应用实践

根据 E-R 图向关系模式的转换原则，将图 1-14 所示的销售业务系统的 E-R 图转换成关系模型。

(1) 图 1-14 中有 4 个实体，加上“账号”实体，转换成 5 个关系，分别为：账号、供应商、商品类型、商品、顾客。

(2) 实体的属性转换为关系的字段，有以下关系模式：

- 账号（用户 ID，密码，邮件，用户名，地址，邮编，电话，状态）。
- 供应商（供应商 ID，名称，地址，邮编，电话，信用状态）。
- 商品类型（类别 ID，类别名称，描述）。
- 商品（商品 ID，名称，价格，产品描述，生产日期，保质期）。
- 顾客（顾客 ID，姓名，性别，年龄，职业，联系方式，地址，办卡时间，积分）。

(3) 将实体的关系转换为关系模式。

1) “供应商”实体和“商品”实体是多对多的关系，需要生成一个新的关系，命名为“进货”，包含的字段有“供应商 ID”、“商品 ID”、“进货时间”、“进货单价”、“进货数量”。

2) “商品类型”和“商品”实体是一对多的关系，将 1 端“商品类型”实体的主键添加到“商品”关系中。

3) “顾客”实体和“商品”实体是多对多的关系，形成一个新的关系，命名为“销售”，包含的字段有“顾客 ID”、“商品 ID”、“数量”、“总价”、“销售时间”。

总的关系模式如下：

- 账号（用户 ID（pk），密码，邮件，用户名，地址，邮编，电话，状态）。
- 供应商（供应商 ID（pk），名称，地址，邮编，电话，信用状态）。
- 商品类型（类别 ID（pk），类别名称，描述）。
- 商品（商品 ID（pk），名称，价格，产品描述，生产日期，保质期，类别 ID（fk））。
- 顾客（顾客 ID（pk），姓名，性别，年龄，职业，联系方式，地址，办卡时间，积分）。
- 进货（供应商 ID（fk），商品 ID（fk），进货时间，进货单价，进货数量），主键为供应商 ID 和商品 ID 的组合。
- 销售（顾客 ID（fk），商品 ID（fk），数量，总价，销售时间），主键为顾客 ID 和商品 ID 的组合。

任务 1.3 规范化数据

1.3.1 情景描述

通过对数据库的分析和设计，得到数据库中有关的关系模式，有的时候会发现有的表存储的冗余度很高，有很多重复的数据，导致我们操作数据的时候出现很多异常。

如果存在关系为：选课（学号，姓名，性别，生日，父母，课程号，课程名，类型，成绩），主键字段是学号和课程号的组合。其中父母字段有父亲和母亲姓名两个值，不符合表的要求；另外如果操作人员发现当一门课程还没有学生选的时候，主键不完整，那么此课程的信息也不能保存；如果学生退学，删除学生的同时，对应的课程信息也全部被删除；发现课程的信息有误，需要修改多次课程的信息。

因此，对存在重复数据的表我们应该如何处理、如何规范？

1.3.2 问题分析

如果一个表中存在很多重复的数据，那么我们发现一个字段有问题的时候修改起来就要每个地方都修改一次。万一失误，可能会造成数据的不一致情况，并且冗余的数据会浪费存储空间。我们需要使用一些规则，将这些带冗余的复杂的表结构拆分成简单表结构来减少冗余。

1.3.3 解决方案

1. 确认关系是否为第一范式（1NF）

对于关系数据库，第一范式是必须遵守的最基本的范式。不满足第一范式的关系不属于表的概念。在 1NF 里，表要求每个单元格的取值是单一的。因此可将以上关系改为：

选课（学号，姓名，性别，生日，父亲，母亲，课程号，课程名，类型，成绩）

2. 确认关系是否为第二范式（2NF）

如果一个关系模式属于第一范式，而且它的任何一个非主属性完全依赖于任一个主关键字，则该关系属于第二范式。比如关系“选课”的主键是“学号”和“课程号”的组合，其中字段“姓名”、“性别”、“生日”、“父亲”、“母亲”只依赖于主键的一部分“学号”字段，与主键的另一部分“课程号”无关；字段“课程名”、“类型”依赖于主键的一部分“课程号”字段，与“学号”无关，那么这样的关系不满足 2NF。

对于以上关系可分别将依赖于部分主键的这些字段消除，重新组成关系学生（学号，姓名，性别，生日，父亲，母亲）和关系课程（课程号，课程名，类型），剩下的关系选课（学号，课程号，成绩），那么这些关系都满足 2NF。

3. 确认关系是否为第三范式（3NF）

如果一个关系模式属于第二范式，并且它的所有非主属性都不传递依赖于主关键字，则该关系满足第三范式。以上学生、课程、选课关系不存在传递依赖的情况，所以满足 3NF。

1.3.4 知识总结

1. 数据库的设计原则

为了获得一个好的数据库，我们要遵循的规则如下：

- 每个表都应该有一个唯一的表名。
- 表中不允许含有多值属性，即每个单元格存储的数据都是一个不可分的数据项。
- 表中任意两行不能有完全相同的数据值，要为表设置主键。
- 表中行与行的顺序任意。
- 表中每列的列名不能重复，但列与列的顺序是任意的。
- 确定每个列的取值范围有效。
- 表中任一列中的值类型必须一致。
- 不能引用一个不存在的数据。

2. 第一范式 (1NF)

一个表结构首先要满足 1NF，要求每个单元格的取值是唯一的，并且是不可再分的数据类型。

表 1-2 不满足 1NF，属性“员工编号”及“员工姓名”的值是一个多值的集合。对这种类型的表格进行操作的时候非常困难，表 1-3 是转换为符合 1NF 的关系。

表 1-2 项目成员报酬表

项目编号	项目名称	员工编号	员工姓名	工作类别	耗时	报酬/小时
PJ01	图书管理信息系统	E0001	岳涵	数据库设计员	20	80
		E0002	露西	程序员	15	50
		E0003	威廉	系统分析员	25	90
PJ02	音乐网站	E0006	史密思	程序员	20	50
		E0009	乔治	测试工程师	15	100
PJ03	网上书店	E0004	琳达	程序员	20	50
		E0008	玛丽	系统分析员	10	90
		E0007	劳瑞	数据库设计员	16	80

表 1-3 符合 1NF 的项目成员报酬表

项目编号	项目名称	员工编号	员工姓名	工作类别	耗时	报酬/小时
PJ01	图书管理信息系统	E0001	岳涵	数据库设计员	20	80
PJ01	图书管理信息系统	E0002	露西	程序员	15	50
PJ01	图书管理信息系统	E0003	威廉	系统分析员	25	90
PJ02	音乐网站	E0006	史密思	程序员	20	50
PJ02	音乐网站	E0009	乔治	测试工程师	15	100
PJ03	网上书店	E0004	琳达	程序员	20	50
PJ03	网上书店	E0008	玛丽	系统分析员	10	90
PJ03	网上书店	E0007	劳瑞	数据库设计员	16	80

3. 第二范式 (2NF)

2NF 在满足 1NF 的基础上，要求行中每个非主属性的取值范围取决于整个主键，而不是取决于主键的一部分。

表 1-3 满足 1NF，但是不满足 2NF，在这个关系中，存在部分依赖于主键的属性。该关系的主键是（项目编号，员工编号），但字段“项目名称”仅由“项目编号”决定，与“员工编号”无关。换言之，属性“项目名称”只是部分依赖于主关键字（项目编号，员工编号），而不是完全依赖。

不满足第二范式（2NF）的表格存在大量冗余，一个项目有多少个员工参与，就会重复多少次，并且在新增、修改、删除记录的时候都会存在如下异常情况：

（1）新增记录：如果一个员工没有分配项目的时候，项目编号为空，主键不完整，那么此员工信息是不能保存的。

（2）修改记录：如果一个项目名称信息输入错误，需要更新，那么这个项目名称出现了多处，就需要修改多个地方，很可能造成数据不一致的情况。

（3）删除记录：如果一个员工离职，需要删除这个员工的信息，刚好这个员工独立完成了一个项目，那么在删除这个员工的同时，与此相对应的项目的信息页被删除了。

综上所述，不满足第二范式的关系会存在插入异常、更新异常、删除异常情况，而一个良好的数据库模式不能存在这样的问题，我们需要重新分配数据项到不同的关系中，来消除这些问题。

首先识别表中的主键、依赖于主键的属性、依赖于主键的一部分的属性；其次对关系进行分解，消除依赖于部分主键的非主属性，把部分依赖的属性单独组建关系。可以把表 1-3 分解为：满足 2NF 的关系在一定程度上解决了冗余、插入异常、更新异常、删除异常。新来的员工还没有分配项目，则可以保存在表 1-4（b）中；项目名称如果写错了，也只需要在表 1-4（c）中修改一次即可；如果一个员工离职了，也只需要在表 1-4（b）中删除，不影响项目的信息。

4. 第三范式（3NF）

3NF 在满足 2NF 的基础上，要求行中每个非主键属性都取决于主键，而不是其他键，即它的任何一个非主属性都不传递依赖于任何关键字。

表 1-4（a）和表 1-4（c）的每个非关键属性都仅仅依赖于主键，不存在传递依赖的情况，满足 3NF。但是表 1-4（b）的“报酬/小时”字段是传递依赖于主键“员工编号”的，这样也存在冗余问题、插入异常、更新异常和删除异常。

表 1-4（a）

项目编号	员工编号	耗时
PJ01	E0001	20
PJ01	E0002	15
PJ01	E0003	25
PJ02	E0006	20
PJ02	E0009	15
PJ03	E0004	20
PJ03	E0008	10
PJ03	E0007	16

表 1-4（b）

员工编号	员工姓名	工作类别	报酬/小时
E0001	岳涵	数据库设计员	80
E0002	露西	程序员	50
E0003	威廉	系统分析员	90
E0006	史密思	程序员	50
E0009	乔治	测试工程师	100
E0004	琳达	程序员	50
E0008	玛丽	系统分析员	90
E0007	劳瑞	数据库设计员	80

表 1-4 (c)

项目编号	项目名称
PJ01	图书管理信息系统
PJ02	音乐网站
PJ03	网上书店

(1) 新增记录: 如果企业还没有招聘到数据库设计员的员工, 那么就没有办法保存这种工作类别的报酬信息。

(2) 修改记录: 如果一种工作类别的报酬需要提高, 那么需要修改多个地方, 必须更新所有属于这个工作类别的员工的报酬信息。

(3) 删除记录: 如果一个员工离职, 需要删除这个员工的信息, 刚好和这个员工的工作类别相同的没有了, 那么在删除这个员工的同时, 与此相对应的工作类别的报酬信息也被删除了。

解决上述问题的方法是: 首先识别传递依赖主键的属性; 其次对关系进行分解, 消除传递依赖主键的属性, 把传递依赖主键的属性组合成新的关系。可以把表 1-4 (b) 分解为表 1-5 (a) 和表 1-5 (b)。

表 1-5 (a)

员工编号	员工姓名	工作类别
E0001	岳涵	数据库设计员
E0002	露西	程序员
E0003	威廉	系统分析员
E0006	史密思	程序员
E0009	乔治	测试工程师
E0004	琳达	程序员
E0008	玛丽	系统分析员
E0007	劳瑞	数据库设计员

表 1-5 (b)

工作类别	报酬/小时
数据库设计员	80
程序员	50
系统分析员	90
测试工程师	100

1.3.5 应用实践

如果存在关系模式销售 (供应商 ID, 供应商名称, 地址, 邮编, 产品 ID, 产品名称, 产品价格, 产品类别, 保质期, 数量), 要求销售关系满足 3NF。

(1) 确定关系销售满足 1NF。

关系中每个单元格的取值都是单一的, 所以满足 1NF。

(2) 确定关系销售满足 2NF。

关系的主键是供应商 ID 和产品 ID 的组合, 存在依赖于主键的一部分的非主属性, 那么不满足 2NF, 找出依赖于部分主键供应商 ID 的非主属性: 供应商名称、地址、邮编, 组合新关系供应商 (供应商 ID, 供应商名称, 地址, 邮编); 找出依赖于部分主键产品 ID 的非主属性: 产品名称、产品类别、保质期, 组合新关系产品 (产品 ID, 产品名称, 产品类别, 保质期), 剩下的字段形成关系销售 (供应商 ID, 产品 ID, 产品价格, 数量)。

(3) 确定关系销售满足 3NF。

供应商（供应商 ID，供应商名称，地址，邮编）、产品（产品 ID，产品名称，产品类别，保质期）、销售（供应商 ID，产品 ID，产品价格，数量）都不存在传递依赖的情况，所以满足 3NF。

单元小结

1. E-R 图表示方法中，实体用矩形表示，关系用菱形表示，属性用椭圆表示。
2. 关系的类型分为一对一、一对多和多对多。
3. E-R 图到表的转换。
4. 关系数据库中，表称为关系，表中的一行称为元组或记录，表中的一列称为字段或者属性，表中列的取值范围称为域。
5. 一个表的主键唯一地识别表中的每一行。
6. 一个表的外键用来表示与另外一个表的关系。
7. 对表实施规范化操作，确保表满足 1NF、2NF 和 3NF。

习题一

1. 什么是数据库？什么是数据库管理系统？
2. 什么是 E-R 模型？构成 E-R 图的基本要素是什么？
3. 根据实体之间的对应关系，可以把关系分为哪几种类型？
4. 什么是关系模型？其表示形式是什么？
5. 什么是主键？什么是外键？
6. 如何把 E-R 图映射为数据库中的表？
7. 数据库的设计原则是什么？
8. 1NF、2NF、3NF 是什么？