

项目 1 设计学生成绩管理数据库

知识目标

- ◆ 掌握数据库的基本概念；
- ◆ 掌握数据库设计方法：概念结构设计、逻辑结构设计和物理结构设计；
- ◆ 掌握关系模型的规范化。

技能目标

- ◆ 熟悉数据库设计的基本步骤；
- ◆ 掌握利用 E-R 图描述数据库的概念模型；
- ◆ 掌握将 E-R 图转化为关系模型的方法；
- ◆ 按照用户的需求设计学生成绩管理数据库。

项目描述

该项目是要完成某校学生成绩管理数据库的设计。

项目分析

项目该项目首先介绍了数据库设计所需的基本概念、设计方法和步骤，然后完成某校学生成绩管理数据库的设计，最后以实例讲解了关系模型的规范化方法。

任务设置

- ◆ 任务 1 认识数据库的历史和发展

- ◆ 任务 2 建立某校“学生成绩管理”数据库模型
- ◆ 任务 3 强化关系数据库的设计过程和原则
- ◆ 任务 4 规范关系数据库

任务 1 认识数据库的历史和发展

任务目标

- 认知数据库。
- 了解数据库的产生和发展。
- 认知数据库的基本概念。

任务描述

学习以后回答以下问题：

- (1) 什么是数据库，它的地位是什么？
- (2) 数据库的发展经历了哪些阶段，每个阶段的特点是什么？
- (3) 什么是信息？什么是数据？什么是数据库管理系统？什么是数据库系统？

1.1 认知数据库

数据库 (Database) 是按照数据结构来组织、存储和管理数据的仓库。数据库技术是信息系统的核心技术，是计算机科学中重要的组成部分，具有高效的数据存取和方便的应用开发等特点。在计算机技术广泛应用的今天，数据库技术的地位也变得越来越重要，它们是电子商务及各种应用程序的主要组成部分，是企业操作和决策的核心部分。

1.2 了解数据库的产生和发展

计算机早期的应用主要是用于科学计算，服务于国防、科学研究等。从 20 世纪 60 年代后期，计算机应用迅速从科学计算进入数据处理阶段，即对各种形式的数据进行搜集、存储、

加工和传播。处理的规模和数据量越来越大，处理技术也不断完善。人们对信息的处理经历了三个阶段：人工管理、文件系统和数据库系统。

1. 人工管理

20 世纪 50 年代中期以前，人们把计算机当做一种计算工具，主要用于科学计算。人们通过编写的程序自带有相关数据，将程序和数据同时录入计算机，当时的计算机没有磁盘，不能对数据进行长期保存。没有操作系统，也没有对数据进行管理的软件，数据是面向程序的，无独立性和不能共享。

2. 文件系统

进入 20 世纪 50 年代后期至 20 世纪 60 年代中期，计算机软硬件都取得进一步发展，计算机产生了以操作系统为核心的系统软件，操作系统提供了文件系统的管理功能。程序和数据可以分别存储为单独的文件，数据文件可以长期存储。这一阶段的数据管理虽然较人工管理有了很多改进，但仍具有不足：数据与程序仍缺乏独立性，数据的冗余度大，数据无结构性，数据存在不一致性。

3. 数据库系统

随着计算机技术的飞速发展，面对信息社会中的大量数据，出现了数据库技术。数据库技术能有效地管理和存取大量的数据，克服了人工管理和文件系统阶段的缺点。数据库系统阶段的特点：数据结构化、数据共享、数据冗余度小，易扩展，程序和数据独立性高，对数据实行了集中和统一的控制。

数据库系统的发展又经历了三个阶段：层次数据库、网状数据库和关系数据库。

关系型数据库已经出现了近 40 年，并且在很长一段时间里一直是数据库领域当之无愧的王者。

随着“大数据时代”的到来，在高并发、大数据量、分布式以及实时性的要求之下，传统的关系型数据库，因为其数据模型以及预定义的操作模式，在很多情况下不能很好满足以上的需求，所以新型数据库如今在大数据的场景下，取代了传统关系型数据库成为主导。相信未来随着大数据的发展，新型数据库将会颠覆数据库领域。

1.3 认知数据库基本概念

1. 数据 (Data)

描述事物的符号记录称为数据。它可以是数字，也可以是文字、图形、图像、声音和语

言等。

2. 信息 (Information)

信息是指经过加工处理的数据，即数据中包含的意义。没有经过加工的数据只是一种原始的材料，它的价值在于记录客观世界的事实。

3. 数据库 (Database , DB)

数据库是存放数据的仓库，即长期存储在计算机内的、有组织、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、描述和存储，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。

4. 数据库管理系统 (Data Base Management System , DBMS)

数据库管理系统是位于用户与操作系统之间的一层数据库管理软件，它的主要作用帮助用户科学的组织和存储数据，高效的获取和维护数据。归纳起来功能是：数据定义、数据操作、数据库的运行和管理、数据库的建立和维护。

5. 数据库系统 (Data Base System , DBS)

数据库系统是指在计算机系统引入数据库后的系统构成，一般由数据库、数据库管理系统 (及开发工具)、应用系统、数据库管理员和用户构成。

在一般不引起混淆的情况下常常把数据库系统简称为数据库。

任务 2 建立某校“学生成绩管理”数据库模型

任务目标

- 理解什么是数据库模型。
- 了解数据模型有哪些层次。
- 认知概念模型，掌握其表示方法。
- 认知关系模型。
- 建立某校“学生成绩管理”数据库模型。

通过本任务，同学们学习以后回答以下问题：

- (1) 什么是数据库模型？
- (2) 数据模型有哪些层次，各个模型有什么作用？
- (3) 怎样表示概念模型？
- (4) 怎样把概念模型转化为关系模型？

2.1 认知数据抽象的层次

模型是现实世界中具体事物的模拟和抽象。例如，一张地图、一架航模飞机，都是具体的模型。数据库是某个企业、组织或部门所涉及数据的综合，它不仅反映数据本身的内容，而且要反映数据之间的联系。由于计算机不可能直接处理现实世界中的具体事物，所以人们必须先要把具体事物转换成计算机能够处理的数据。

在数据库中，用数据模型来抽象、表示和处理现实世界中的数据和信息。根据数据抽象层次，针对不同的数据对象和应用目的，可以将数据模型分为概念数据模型、逻辑数据模型和物理数据模型三类。

2.1.1 概念数据模型

概念数据模型是独立于计算机系统的数据模型，它完全不涉及信息在计算机系统中的应用，只是用来描述所使用的信息结构。概念模型是现实世界的第一层抽象，强调语义表达能力，是用户和数据库设计人员之间进行交流的工具。概念模型简单、清晰，易于用户理解。

2.1.2 逻辑数据模型

逻辑数据模型是现实世界的第二层抽象，反映了数据的逻辑结构，例如文件、记录和字段等。它是某个数据库管理系统所支持的逻辑数据模型，主要有层次数据模型、网状数据模型和关系数据模型。逻辑数据模型一方面要面向用户，使用户能够看到数据库的数据模型；另一方面又要通过严格的形式化定义，以便在计算机上实现。

2.1.3 物理数据模型

物理数据模型反映数据在计算机中的存储结构，例如，存储介质的物理块、指针和索引等。每个逻辑数据模型在实现时，都有其对应的物理数据模型。物理数据模型不但与具体的数据库管理系统有关，而且还与计算机系统的硬件和操作系统有关。

2.2 认知概念模型

具体的数据库管理系统所支持的逻辑数据模型不便于非计算机专业人员理解和应用，在开始设计数据模型时，可以先用概念模型将现实世界中的客观事物用某种信息结构表示出来，再转化为用计算机表示的逻辑数据模型，如图 1-1 所示。

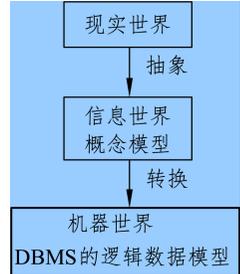


图 1-1 数据抽象层次

概念模型不涉及信息在计算机系统中的应用，是面向现实世界的第一层抽象，主要用于按照用户的观点来对数据和信息进行建模。作为现实世界和机器世界的一个中间层次，概念模型是数据库设计人员进行数据库设计的工具与用户进行交流的语言。

2.2.1 认知实体及其属性

1. 实 体

现实世界中客观存在并可相互区分的事物称为实体。实体可以是具体的人、事、物，也可以是抽象的概念，例如，一名学生、一个部门、一门课、一个规划等都是实体。

2. 属 性

实体所具有的某一特性称为属性。一个实体可以由若干个属性来描述。例如，学生实体可以由学号、姓名、性别、出生日期、所在系等属性组成（1101，王立明，男，19841020，计算机系）。这些属性组合起来描述了一个学生实体。

对于一个对象，根据处理问题的需要可选择不同的属性描述不同实体。例如，一个学生，在学籍管理中用姓名、年龄、性别、籍贯、家庭住址等属性作为一个实体；而在成绩管理中可用课程名称、学分、成绩等属性作为另一个实体。

3. 实体和属性的型与值

型是结构，用实体名及其属性名集合描述同类实体，称为实体型；值是数据，不同的实体有不同的属性内容（属性值）。

4. 实体集

具有相同实体型的实体值的集合称为实体集。

5. 关键字

在实体属性中，能区别实体集合中不同个体的某一个或某几个属性的组合，称为关键字。

2.2.2 认知实体间的联系

在现实世界中，事物之间是有联系的，在信息世界中反映为实体之间的关系。两个实体间的关系通常指一个实体集中各实体间的关系或者不同实体集中实体之间的关系。

两个实体集之间的联系可以分为三类：

(1) 一对一联系 (1:1)。

如果实体集 A 中的每个实体与实体集 B 中最多一个实体有关系，反之实体 B 中的每个实体与实体集 A 中至多一个实体有关系，则称实体集 A 与实体集 B 是一对一联系，记为 1:1。

例如，学校里约定一个学院只有 1 个院长，而一个院长只能在一个学院任职，则学院与院长两个实体集之间具有一对一联系。

(2) 一对多联系 (1:N)。

如果实体集 A 中的每个实体与实体集 B 中任意多个实体有关系，反之实体 B 中的每个实体与实体集 A 中至多一个实体有关系，则称实体集 A 与实体集 B 是一对多联系，记为 1:N。

例如，学院中有若干名教师，而一个教师只属于一个学院，则学院与教师两个实体集之间具有一对多联系。

(3) 多对多联系 (M:N)。

如果实体集 A 中的每个实体与实体集 B 中任意多个实体有关系，反之实体集 B 中的每个实体也与实体集 A 中任意多个实体有关系，则称实体集 A 与实体集 B 是多对多联系，记为 M:N。

例如，一门课同时有若干名学生学习，而一个学生可以同时学习多门课程，则课程与学生两个实体集之间具有多对多联系。

两个实体集之间的以上三种关系示例如图 1-2 所示。

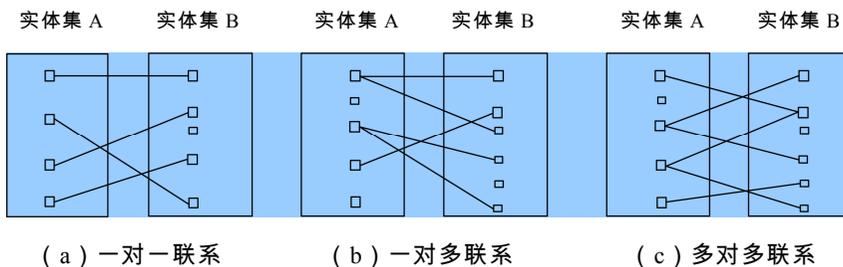


图 1-2 体集之间的联系类型

2.2.3 概念模型

1. 方法

概念模型的表示方法很多，其中最著名、最常用的是 P. P. S. Chen 于 1976 年提出的实体-联系方法 (E-R 方法)。该方法用 E-R 图描述信息世界的概念模型。方法如下：

① 实体型：用矩形表示，矩形框内写上实体名。

② 属性：用椭圆形或圆角矩形表示，图形内写上属性名，并用直线将其与相应的实体连接起来。

③ 联系：用菱形表示，菱形框内写上表示联系行为的动词以表示联系名，并用直线分别与有关实体连接起来，同时在直线旁边标上联系的类型 (1:1, 1:N 或 M:N)。如果一个联系具有属性，则这些属性也要用直线与该联系连接起来。

2. 示例

“学生”实体及属性的画法。

通常在表示单个实体的 E-R 图中，除了要标明实体名外，还要标明实体所具有的属性。例如，学生实体具有学号、姓名、性别、出生日期和所在系的属性，其 E-R 图如图 1-3 所示。

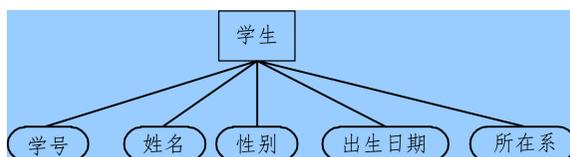
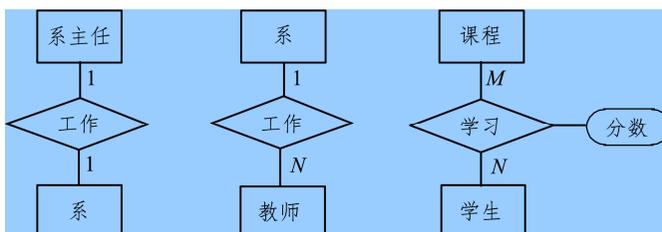


图 1-3 “学生”实体及属性的表示

实体之间的联系画法。

在表示实体之间的联系时，每个实体只标明其实体名，而把实体属性单独用图表示或把所有实体集用列表表示。

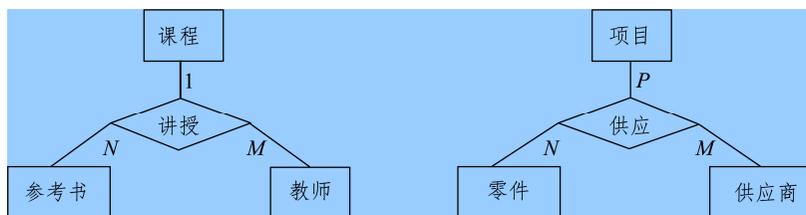
图 1-4 表示了两个实体集的 E-R 图。其中课程和学生实体集之间是多对多联系，并且关系具有属性“分数”。



(a) 1:1 联系 (b) 1:N 联系 (c) M:N 联系

图 1.4 两个实体集联系的表示

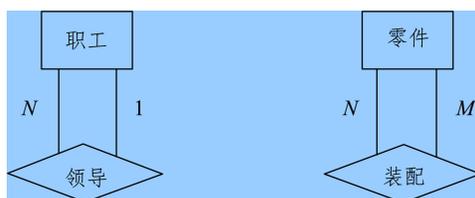
图 1.5 表示了三个实体集的 E-R 图。



(a) 课程与参考书、教师之间 1:N 联系 (b) 项目、零件和供应商之间 M:N 联系

图 1-5 三个实体集的 E-R 图表示

同一个实体集内部联系的表示，如图 1-6 所示。



(a) 1:N 联系 (b) M:N 联系

图 1-6 相同实体集内部联系的表示

课堂练习

设有商店和顾客两个实体。“商店”有属性：商店编号、商店名、地址、电话。“顾客”有属性：顾客编号、姓名、地址、年龄、性别。假设一个商店有多个顾客购物，一个顾客可以到多个商店购物，顾客每一次去商店购物有一个消费金额和日期，试画出 E-R 图，并注明属性和联系类型。

建立步骤：

(1) 确定实体及属性。本题已知有两个实体，其中商店实体图如图 1-7 所示。

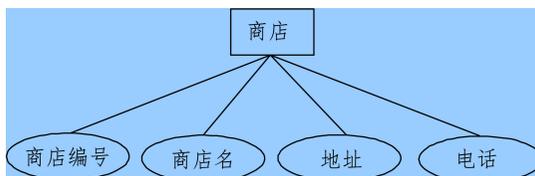


图 1-7 商店实体图

顾客实体图如图 1-8 所示。

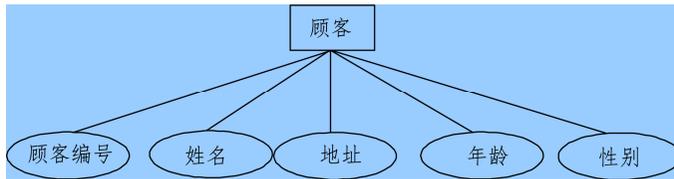


图 1-8 顾客实体图

(2) 确定联系及属性。

- “商店-顾客”的联系是 $m:n$ 。
- 联系的属性有四项：顾客（编号）、商店（编号）、消费金额、日期。
- 联系的名称：购物。

示意图如图 1-9 所示。

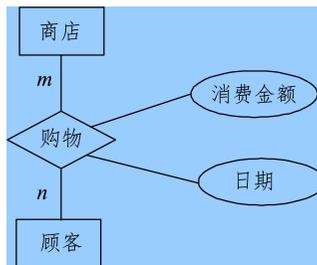


图 1-9 购物联系

(3) 把实体和联系组合成 E-R 图。

商店-顾客实体联系如图 1-10 所示。

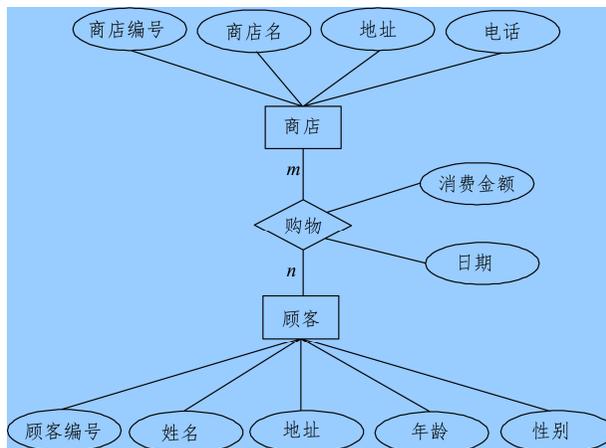


图 1-10 商店-顾客实体联系图

2.3 认知关系模型

数据库的概念模型按出现的时间先后有层次模型、网状模型、关系模型、面向对象模型。关系型数据库采用的概念模型是关系模型。

2.3.1 认知关系模型的基本概念

关系模型是当前使用最广泛的数据模型。关系数据模型是以集合论中的关系概念为基础逐步发展起来的。自 20 世纪 80 年代以来，计算机厂商新推出的数据库产品几乎都支持关系模型，非关系系统的产品也大都加上了关系接口。数据库领域当前的研究工作都是以关系方法为基础的。Microsoft SQL Server 2008 数据库管理系统也是基于关系模型的。关系模型是建立在数学概念基础上的，它的主要特征是使用关系来表示实体以及实体之间的联系。

1. 关系

一个关系模型的逻辑结构是二维表，它由行和列组成。一个关系对应一张二维表，用于存储数据，表中的每一行代表一个实体，表中的每一列都用来描述实体的特征。每一个关系都有一个名字，如“学生”关系如表 1-1 所示。

表 1-1 “学生”关系

学号	姓名	性别	出生日期	所在系
1001	吕川页	男	1985-3-5	信息
1002	郑学敏	女	1984-4-16	信息
1003	于丽	男	1985-8-4	数学
1004	孙立华	女	1986-7-3	数学
...

2. 元组

表中的一行称为一个元组，在数据库中也称为记录，在“学生”关系中 (1001, 吕川页, 男, 1985-3-5, 信息) 就是一个元组，表示一个特定的学生记录。

3. 属性

表中的一列称为一个属性，用来描述事物的特征，属性分为属性名和属性值。例如，学生的姓名、学号、性别等为属性名，其对应的具体取值为属性值。在数据库中属性也称为字段。

4. 域

域即属性的取值范围。如“学生”关系中出生日期的域是合法的日期，学号的域是若干位数字组成的字符集合，性别的域是{男，女}。在数据库中属性的取值由其数据类型决定。

5. 关系模式

关系的描述称为关系模式(Relation Schema)。关系模式通常可以简记为 $R(A_1, A_2, \dots, A_n)$ 。其中 R 为关系名， A_1, A_2, \dots, A_n 为属性名。而域名及属性向域的映像常常直接说明为属性的类型、长度。关系实际上就是关系模式在某一时刻的状态或内容。也就是说，关系模式是型，关系是它的值。关系模式是静态的、稳定的，而关系是动态的、随时间不断变化的，因为关系操作在不断地更新着数据库中的数据。

6. 关键字/码

若关系中的某一个属性或属性组的值唯一地决定其他所有属性，则这个属性或属性组称为该关系的关键字。在“学生”关系中，学号就是关键字，所有学生的学号是唯一的。而在“学习”关系中，只有学号和课程号的组合属性才可以作为该关系的关键字。

7. 候选键/候选关键字/候选码

如果一个关系中有多个属性或属性组都能用来标识该关系的元组，那么这些属性或属性组都称为该关系的候选关键字。如“课程”关系中，课程号可以作为关键字，若课程名称能够保证没有重复，则也可以用课程名称作为关键字，那么，课程号和课程名都是该关系的候选关键字。

8. 主键/主关键字/主码

在一个关系的多个候选关键字中指定其中一个作为该关系的关键字，则称它为主关键字或主键、主码。如，“课程”关系中，若课程名能保证没有重复，则既可以选择课程号作为主键，也可以选择课程名作为主键。

9. 外键/外关键字/外码

如果一个关系 R 中的某个属性或属性组 F 并非该关系的关键字，但它和另外一个关系 S 的关键字 K 相对应，则称 F 为关系 R 的外键，同时要求外键 F 的值要参照关系 S 中主键 K 的值。如“学习成绩”关系中的“学号”属性不能作为该关系的主键，它的值要引用“学生”关系的主键学号，则“学习成绩”关系中的“学号”属性称为该关系的外键。在这里，我们就把 S 表称为主表，R 表称为从表。

2.3.2 认知 E-R 模型与关系模型的转换方法

E-R 模型必须要转化成关系模型，转换原则叙述如下。

1. 实体向关系模型的转化

每个实体转换为一个关系(数据表格),实体的属性就是关系的属性。实体的关键字作为关系的关键字,根据实际对象属性情况确定关系属性的值域。

2. 一对一关系向关系模型的转化

1:1 的联系既可以转换为一个独立的关系,也可以与任意一端实体对应的关系合并。

① 转化为一个独立的关系。

关系的属性为与该联系相连的各个实体的主键以及联系本身的属性,可以选择任意一端实体的主键作为该关系的主键。

② 与任意一端实体对应的关系合并。

选择任意一端实体,在其对应的关系中加入另一端实体的主键和联系本身的属性。

3. 一对多关系向关系模型的转化

1:N 的联系既可以转换为一个独立的关系,也可以与 N 端实体对应的关系合并。

① 转换为 1 个独立的关系。

关系的属性为与该联系相连的各实体的主键以及联系本身的属性,N 端实体的主键可以作为该关系的主键。

② 与 N 端对应的关系合并。

与 N 端对应的关系中加入一端实体的主键和联系本身的属性。

4. 多对多联系向关系模型的转化

一个 M:N 联系要转换为一个独立的关系。关系的属性为与该联系相连的各实体的主键以及联系本身的属性,关系的主键为各实体的主键的组合。

2.4 综合任务——设计“学生成绩管理”数据库模型

某校的学生成绩管理信息系统是一个用来管理学生、课程、教师以及学生选修课程后成绩的数据库系统。

学生成绩管理的主要业务流程描述是:学生选修指导老师课程,教师上课,学生通过学习

获得该课程的成绩和学分，规定一个老师可以教多个学生课程，一个学生只有一个指导老师。

(1) 根据上述语义画出 E-R 图。

(2) E-R 图转换成关系数据模型，并指出每个关系模式的主键和外键。

解答：

(1) 建立 E-R 图。

① 确定实体类型及属性。

学生 (学号, 姓名, 性别, 年龄, 籍贯, 专业)

课程 (课程号, 课程名, 学时数, 学分, 教材)

教师 (教师号, 姓名, 性别, 年龄, 专业, 职称)

② 确定联系及类型。

教师-课程的联系 (1:N) —— 授课。

学生-课程的联系 (M:N) —— 选修。属性有：成绩。

教师-学生的联系 (1:N) —— 指导。属性有：指导时间、地点。

③ 把实体类型和联系类型组合成 E-R 图，如图 1-11 所示。

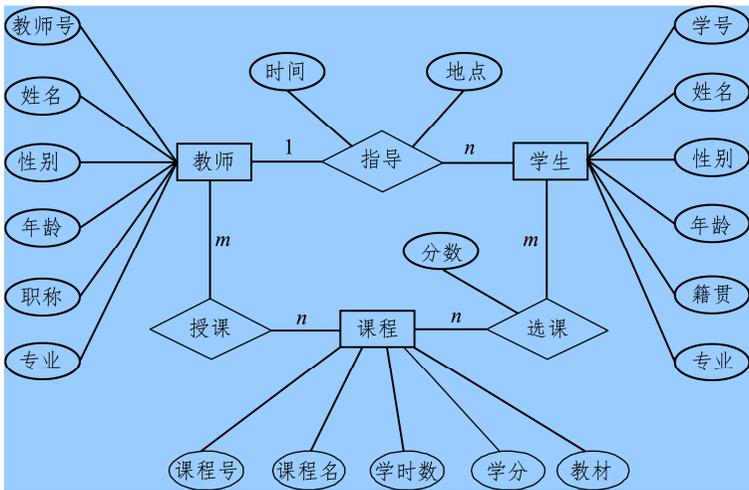


图 1-11 教学管理 E-R 图

(2) 转化为关系模式。

- ◆ 教师 (教师号, 姓名, 性别, 年龄, 职称, 专业), 主关键字：教师号
- ◆ 学生 (学号, 姓名, 性别, 年龄, 籍贯, 专业), 主关键字：学号
- ◆ 课程 (课程号, 课程名, 学时数, 学分, 教材), 主关键字：课程号
- ◆ 选修 (学号, 课程号, 成绩), 主关键字：学号 + 课程号, 外关键字：学号、课程号
- ◆ 授课 (教师号, 课程号), 主关键字：教师号 + 课程号, 外关键字：教师号、课程号
- ◆ 指导 (学号, 教师号, 时间, 地点), 主关键字：学号, 外关键字：教师号

(3) 合并相同码的关系模式。

由于学生关系和指导关系的关键字都是“学号”，所以可以把这两个关系合并为一个。合并后的关系是：

- ◆ 学生 (学号 , 姓名 , 性别 , 年龄 , 专业 , 指导教师号 , 时间 , 地点) , 主关键字 : 学号。