

## 考点突破



根据考试大纲, 本章要求考生掌握以下几个方面的知识点。

- ( 1 ) 数据库模型 ( 概念模式、外模式、内模式 )
- ( 2 ) 数据模型, ER图, 规范化
- ( 3 ) 数据操作
- ( 4 ) 数据库语言
- ( 5 ) 数据库管理系统的功能和特征
- ( 6 ) 数据库的控制功能
- ( 7 ) 数据仓库和分布式数据库基础知识

从历年的考试情况来看, 本章的考点主要集中在 : E-R模型、关系代数、元组演算、规范化理论 ( 键、范式、模式分解 ) 、SQL语言等。

[版权方授权希赛网发布, 侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

## 数据库模式及ER模型

数据库是长期存储在计算机内的、有组织的、可共享的数据集合, 数据库系统是指在计算机信息系统中引入数据库后的系统, 一般由数据库、数据库管理系统 ( DataBase Management System, DBMS ) 、应用系统、数据库管理员 ( DataBase Administrator, DBA ) 和用户构成。数据库系统的结构可以有多种不同的层次或不同的角度, 其中典型的是三级划分法, 其中包括三级模式和两级映射。下面将就该主题, 以及ER模型展开论述。

[版权方授权希赛网发布, 侵权必究](#)

[上一节](#) [本书简介](#) [下一节](#)

## 考点精讲

### 1. 三级模式与两级映射

数据库系统由外模式、概念模式和内模式三级构成。其关系如图5-1所示。

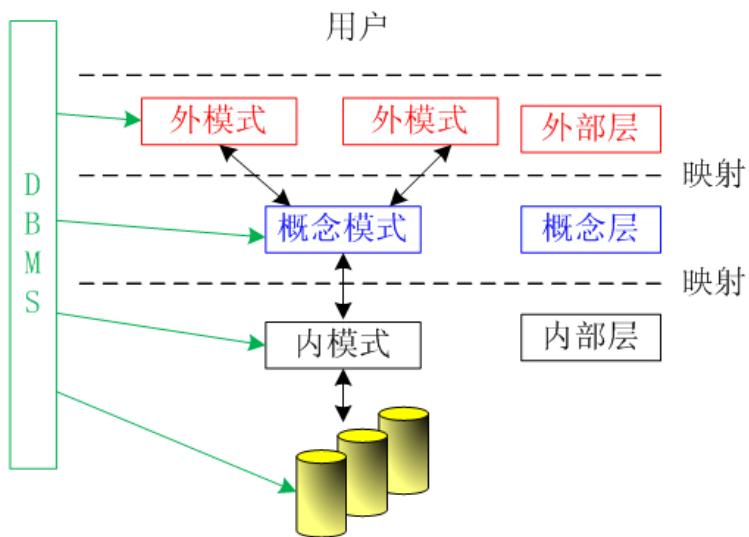


图5-1 数据库三级模式两级映射示意图

从图5-1可以看出，整个数据库体系由数据库管理系统（DBMS）进行管理，其内容涉及底层的数据存储问题，顶层涉及与用户的交互，这种层次的划分，主要目标是使数据库体系内部耦合度更低，变得更为灵活。

外模式也称为子模式或用户模式，它对应的是我们平时所用到的数据库视图。外模式用来描述用户（包括程序员和最终用户）看到或使用的那部分数据的逻辑结构，是数据库用户的数据视图，是与某一应用有关的数据的逻辑表示。一个数据库可以有多个外模式，一个应用程序只能使用一个外模式。

概念模式也称为模式或逻辑模式，它对应我们平时所用到的数据表。概念模式是数据库中全体数据的逻辑结构和特征的描述，是所有用户的公共数据视图，用以描述现实世界中的实体及其性质与联系，定义记录、数据项、数据的完整性约束条件及记录之间的联系。概念模式通常还包含有访问控制、保密定义和完整性检查等方面的内容，以及概念/物理之间的映射。一个数据库只有一个概念模式。

内模式对应于物理级数据库，是数据物理结构和存储方式的描述，是数据在数据库内部的表示方式。内模式不同于物理层，它假设外存是一个无限的线性地址空间。内模式定义的是存储记录的类型、存储域的表示和存储记录的物理顺序，以及索引和存储路径等数据的存储组织。一个数据库只有一个内模式。

在数据库系统的三级模式中，模式是数据库的中心与关键；内模式依赖于模式，独立于外模式和存储设备；外模式面向具体的应用，独立于内模式和存储设备；应用程序依赖于外模式，独立于模式和内模式。

从图5-1可以看出，三级模式实际对应着数据库系统中的三个层次，这三个层次划分出来以后，为了达到“一个层次的变化不对其它两个层产生影响”的效果，提出了两级映射。两级映射分别是：外模式与概念模式之间的映射、概念模式与内模式之间的映射。

外模式与概念模式之间的映射：用于维护数据库的逻辑独立性。也就是说，有了这个映射，使得数据的逻辑结构改变时，应用程序不需要改变，只需要改变映射中的对应关系即可达到目的。

概念模式与内模式之间的映射：用于维护数据库的物理独立性。也就是说当数据的物理存储改变时，应用程序不需要改变。

## 2. ER模型

数据库系统是对现实世界中数据的一种抽象，正如在“数据库系统功能和特性”知识点中所描述的，首先我们将通过概念模型将现实世界抽象成为信息世界，然后再抽象成为基本数据模型。而最常使用的概念模型就是E-R模型，本知识点则主要围绕着这个重要概念阐述。

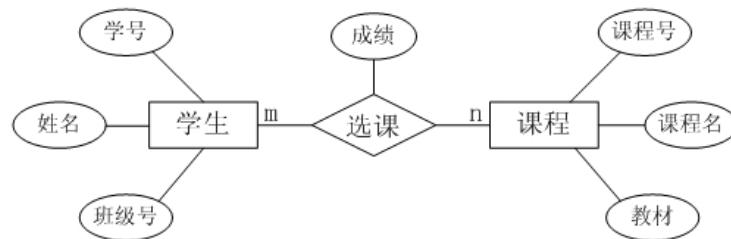


图5-2ER模型

图5-2展示了一个简单的ER模型，我们通过该图进行ER模型的概念入门：

实体：客观存在并可相互区别的事物，可以是具体的人、事、物，也可以是抽象的概念或联系。图5-2中的“学生”与“课程”便是实体。

属性：实体所具有的某一特性称为属性，通常一个实体可以由多个属性来描述。图5-2中“学生”实体旁边的“学号”、“姓名”、“班级号”便是属性。

联系：实体内部的联系通常是指组成实体的各属性间的关系。图5-2中“选课”便是联系。

### (1) ER模型实体联系类型

一对联系 (1:1)：对于实体集A中的每一个实体，实体集B中至多有一个实体与之联系。例：一个班级只有一个班主任，一个班主任也只在一个班级中任职。

一对多联系 (1:n)：对于实体集A中的每一个实体，实体集B中有n (n>0) 个实体与之联系。反之，实体集B中的每一个实体，实体集A中至多只有一个实体与之联系。例：一个班级中有许多学生，而每个学生只在一个班级中学习。

多对多联系 (m:n)：对于实体集A中的每一个实体，实体集B中有n (n>0) 个实体与之联系。反之，实体集B中的每一个实体，实体集A中有m (m>0) 个实体与之联系。一门课程同时有许多学生选修，而一个学生也可以选修多门课程。如图5-2所示，该ER模型便属于m:n型。

### (2) E-R模型的集成

在数据库的概念设计过程中，由于系统都存在一定的复杂度，一次性设计全局E-R图将存在较大风险，所以一般会先设计各子系统的局部E-R图，然后进行集成。

但由于各子系统应用所面临的问题不同，且通常是由不同的设计人员进行局部视图设计，这就导致各个局部E-R图之间必定会存在许多不一致的问题，称之为冲突。因此，在合并E-R图时，不能简单地将各个局部E-R图画到一起，而是必须着力消除各个局部E-R图中的不一致，以形成一个能为全系统中所有用户共同理解和接受的统一的概念模型。各局部E-R图之间的冲突主要有三类，分别是属性冲突、命名冲突和结构冲突。

1) 属性冲突。属性冲突包括属性域冲突和属性取值冲突。属性冲突理论上好解决，只要换成相同的属性就可以了，但实际上需要各部门协商，解决起来并不简单。

2) 命名冲突。命名冲突包括同名异义和异名同义。处理命名冲突通常也像处理属性冲突一样，通过讨论和协商等行政手段加以解决。

3) 结构冲突。结构冲突包括同一对象在不同应用中具有不同的抽象，以及同一实体在不同局部E-R图中所包含的属性个数和属性排列次序不完全相同。对于前者的解决办法是将属性变换为实体或实体变换为属性，使同一对象具有相同的抽象。对于后者的解决办法是使该实体的属性取各局部E-R

图中属性的并集，再适当调整属性的次序。

### (3) E-R模型转关系模式

E-R图向关系模式的转换属于数据库的逻辑设计阶段的工作，该阶段需要将E-R模型转换为某种DBMS能处理的关系模式，具体转换规则如下：

- 1) 一个实体转换为一个关系模式，实体的属性就是关系的属性，实体的主键就是关系的主键。
- 2) 一个1:1联系可以转换为一个独立的关系模式，也可以与任意一端对应的关系模式合并。如果转换为一个独立的模式，则与该联系相连的各实体的主键和联系本身的属性均转换为关系的属性，每个实体的主键均是该关系的键属性；如果与某一端实体对应的关系模式合并，则需要在该关系模式的属性中加入另一个关系模式的主键和联系本身的属性。
- 3) 一个1:n联系可以转换为一个独立的关系模式，也可以与任意n端对应的关系模式合并。如果转换为一个独立的模式，则与该联系相连的各实体的主键和联系本身的属性均转换为关系的属性，而关系的主键为n端实体的主键；如果与n端实体对应的关系模式合并，则需要在该关系模式的属性中加入1端关系模式的主键和联系本身的属性。
- 4) 一个m:n联系转换为一个独立的关系模式，与该联系相连的各实体的主键以及联系本身的属性均转换为关系的属性，而关系的主键为各实体主键的组合。
- 5) 三个以上实体间的一个多元联系可以转换为一个独立的关系模式，与该联系相连的各实体的主键和联系本身的属性均转换为关系的属性，而关系的主键为各实体主键的组合。

另外，还有三种情况是需要特别注意的：

- 1) 多值属性的处理。如果E-R图中某实体具有一个多值属性，则应该进行优化，把该属性提升为一个实体，通常称为弱实体；或者在转化为关系模式时，将实体的主键与多值属性单独构成一个关系模式。
- 2) 派生属性的处理。因为派生属性可由其他属性计算得到，因此，在转化成关系模式时，通常不转换派生属性。
- 3) 在面向对象模型中，关系模式就对应类，关系模式的属性就对应类的属性。

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第5章：数据库系统基础知识

作者：希赛教育软考学院 来源：希赛网 2014年05月04日

## 一点一练

### 试题1

E-R模型向关系模型转换时，三个实体之间多对多的联系m:n:p应该转换为一个独立的关系模式，且该关系模式的关键字由\_\_(1)\_\_组成。

- (1) A. 多对多联系的属性 B. 三个实体的关键字
- C. 任意一个实体的关键字 D. 任意两个实体的关键字

### 试题2

在某企业的营销管理系统设计阶段，属性“员工”在考勤管理子系统中被称为“员工”，而在档案管理子系统中被称为“职工”，这类冲突称为\_\_(2)\_\_冲突。

- (2) A. 语义    B. 结构    C. 属性    D. 命名

### 试题3

E-R图转换为关系模型时，对于实体E<sub>1</sub>与E<sub>2</sub>间的多对多联系，应该将\_\_(3)\_\_\_。

- (3) A. E<sub>1</sub>的码加上联系上的属性并入E<sub>2</sub>  
B. E<sub>1</sub>的码加上联系上的属性独立构成一个关系模式  
C. E<sub>2</sub>的码加上联系上的属性独立构成一个关系模式  
D. E<sub>1</sub>与E<sub>2</sub>码加上联系上的属性独立构成一个关系模式

版权方授权希赛网发布，侵权必究

[上一节](#)    [本书简介](#)    [下一节](#)

第5章：数据库系统基础知识

作者：希赛教育软考学院    来源：希赛网    2014年05月04日

## 解析与答案

### 试题1分析

在E-R模型向关系模型转换时，如果是多对多的联系，那么这个联系需要转换为一个独立的关系模式，且该关系模式的属性由各实体的关键字和该联系自身的属性组成，而该关系模式的关键字（主键）由各实体的关键字组成。

### 试题1答案

- (1) B

### 试题2分析

在本题中显然是存在命名冲突，同一个实体在同系统中存在不同的命名，这是不被允许的。

### 试题2答案

- (2) D

### 试题3分析

本题主要考查E-R图转换为关系模式的规则。

在多对多联系转换为关系模式时，应该将两个实体分别转换为一个关系模式，并将其联系也转换为一个独立的关系模式，该关系模式的属性包括联系本身的属性即两实体的主键。因此本题的答案选D。

### 试题3答案

- (3) D

版权方授权希赛网发布，侵权必究

[上一节](#)    [本书简介](#)    [下一节](#)

第5章：数据库系统基础知识

作者：希赛教育软考学院    来源：希赛网    2014年05月04日

## 考点精讲

## 1. 关系运算

关系代数的基本运算主要有并、交、差、笛卡尔积、选择、投影、连接和除法运算。

(1) 并。计算两个关系在集合理论上的并集，即给出关系R和S（两者有相同元/列数）， $R \cup S$ 的元组包括R和S所有元组的集合，形式定义如下：

$$R \cup S \equiv \{t \mid t \in R \vee t \in S\}$$

式中t是元组变量（下同）。显然， $R \cup S = S \cup R$ 。

(2) 差。计算两个关系区别的集合，即给出关系R和S（两者有相同元/列数）， $R - S$ 的元组包括R中有而S中没有的元组的集合，形式定义如下：

$$R - S \equiv \{t \mid t \in R \wedge t \notin S\}$$

(3) 交。计算两个关系集合理论上的交集，即给出关系R和S（两者有相同元/列数）， $R \cap S$ 的元组包括R和S相同元组的集合，形式定义如下：

$$R \cap S \equiv \{t \mid t \in R \wedge t \in S\}$$

显然， $R \cap S = R - (R - S)$ 和 $R \cap S = S - (S - R)$ 成立。

(4) 笛卡尔积。计算两个关系的笛卡尔乘积，令R为有m元的关系，S为有n元的关系，则 $R \times S$ 是m+n元的元组的集合，其前m个元素来自R的一个元组，而后n个元素来自S的一个元组。形成定义如下：

$$R \times S \equiv \{t \mid t = \langle t_r, t_s \rangle \wedge t_r \in R \wedge t_s \in S\}$$

若R有u个元组，S有v个元组，则 $R \times S$ 有 $u \times v$ 个元组。

(5) 投影。从一个关系中抽取指明的属性（列）。令R为一个包含属性A的关系，则

$$\pi_A(R) \equiv \{t[A] \mid t \in R\}$$

(6) 选择。从关系R中抽取出满足给定限制条件的记录，记作：

$$\sigma_F(R) \equiv \{t \mid t \in R \wedge F(t) = \text{true}\}$$

其中F表示选择条件，是一个逻辑表达式（逻辑运算符+算术表达式）。选择运算是从元组（行）的角度进行的运算。

(7) θ连接。θ连接从两个关系的笛卡尔积中选取属性之间满足一定条件的元组，记作：

$$R \bowtie_{A \theta B} S \equiv \{t, t_s \mid t_r \in R \wedge t_s \in S \wedge t_r[A] \theta t_s[B]\}$$

其中A和B分别为R和S上元数相等且可比的属性组。θ为“=”的连接，称为等值连接，记作：

$$R \bowtie_{A=B} S \equiv \{t, t_s \mid t_r \in R \wedge t_s \in S \wedge t_r[A] = t_s[B]\}$$

如果两个关系中进行比较的分量必须是相同的属性组，并且在结果中将重复的属性去掉，则称为自然连接，记作：

$$R \bowtie S \equiv \{t, t_s \mid t_r \in R \wedge t_s \in S \wedge t_r[A] = t_s[B]\}$$

(8) 除。设有关系R(X, Y)与关系S(Z), Y和Z具有相同的属性个数，且对应属性出自相同域。关系R(X, Y)÷S(Z)所得的商关系是关系R在属性X上投影的一个子集，该子集和S(Z)的笛卡尔积必须包含在R(X, Y)中，记为R÷S，其具体计算公式为：

$$R \div S = \pi_{1, 2, \dots, r-s}(R) - \pi_{1, 2, \dots, r-s}((\pi_{1, 2, \dots, r-s}(R) \times S) - R)$$

例如，有关系R与关系S如表5-1和表5-2所示。

表5-1 关系R

U1	U2	U3	U4
a	b	c	d
a	b	e	f
c	a	c	d

表5-2 关系S

U3	U4
c	d
e	f

则 $R \div S$ 的求解过程为：首先，按除运算定义要求，确定 $X, Y, Z$ 属性集合。 $Y$ 是关系 $R$ 中的属性集合， $Z$ 是 $S$ 中全部属性的集合，即 $Z=\{U3, U4\}$ ，由于 $Y=Z$ ，因此， $Y=\{U3, U4\}$ ， $X=\{U1, U2\}$ 。也就是说， $R \div S$ 结果集包含属性 $U1$ 和 $U2$ ；然后，将关系 $R$ 的 $U1, U2$ （共有 $\langle a, b \rangle, \langle c, a \rangle$ 两个元组）与关系 $S$ 作笛卡尔积操作，结果如表5-3所示。

表5-3  $R(U1, U2) \times S$

U1	U2	U3	U4
a	b	c	d
a	b	e	f
c	a	c	d
c	a	e	f

通过检查表5-3，可以发现元组 $\langle a, b \rangle$ 与 $S(Z)$ 的笛卡尔积被包含在 $R(X, Y)$ 中，而元组 $\langle c, a \rangle$ 与 $S(Z)$ 的笛卡尔积有一个元组未被包含在 $R(X, Y)$ 中，所以，结果集中只有元组 $\langle a, b \rangle$ 。

## 2. 元组演算

在元组演算中，元组演算表达式简称为元组表达式，其一般形式为 $\{t|P(t)\}$ ，其中， $t$ 是元组变量，表示一个元数固定的元组； $P$ 是公式，在数理逻辑中也称为谓词，也就是计算机语言中的条件表达式。 $\{t|P(t)\}$ 表示满足公式 $P$ 的所有元组 $t$ 的集合。

在元组表达式中，公式由原子公式组成，原子公式有下列两种形式：

(1)  $R(s)$ ，其中 $R$ 是关系名， $s$ 是元组变量。其含义是“ $s$ 是关系 $R$ 的一个元组”。

(2)  $s[i]\theta u[j]$ ，其中 $s$ 和 $u$ 是元组变量， $\theta$ 是算术比较运算符， $s[i]$ 和 $u[j]$ 分别是 $s$ 的第 $i$ 个分量和 $u$ 的第 $j$ 个分量。原子公式 $s[i]\theta u[j]$ 表示“元组 $s$ 的第 $i$ 个分量与元组 $u$ 的第 $j$ 个分量之间满足 $\theta$ 运算”。例如，“ $t[2] < u[3]$ ”表示元组 $t$ 的第2个分量小于元组 $u$ 的第3个分量。这个原子公式的一种简化形式是 $s[i]\theta a$ 或 $a\theta u[j]$ ，其中 $a$ 为常量。例如，“ $t[4]=3$ ”表示 $t$ 的第4个分量等于3。

在一个公式中，如果元组变量未用存在量词“ $\exists$ ”或全称量词“ $\forall$ ”等符号定义，那么称为自由元组变量，否则称为约束元组变量。公式的递归定义如下。

(1) 每个原子是一个公式，其中的元组变量是自由变量。

(2) 如果 $P1$ 和 $P2$ 是公式，那么， $\neg P1, P1 \vee P2, P1 \wedge P2$ 和 $P1 \rightarrow P2$ 也是公式。

(3) 如果 $P1$ 是公式，那么 $(\exists s)(P1)$ 和 $(\forall s)(P1)$ 也都是公式。

(4) 公式中各种运算符的优先级从高到低依次为 $\theta$ 、 $\exists$ 和 $\forall$ 、 $\neg$ 、 $\wedge$ 和 $\vee$ 、 $\rightarrow$ 。在公式外还可以加括号，以改变上述优先顺序。

(5) 公式只能由上述四种形式构成，除此之外构成的都不是公式。

在元组演算的公式中，有下列四个等价的转换规则：

(1)  $P1 \wedge P2$ 等价于 $\neg(\neg P1 \vee \neg P2)$ 。

(2)  $P1 \vee P2$ 等价于 $\neg(\neg P1 \wedge \neg P2)$ 。

(3) ( $\forall s$ )( $P_1(s)$ )等价于  $\neg(\exists s)(\neg P_1(s))$  ; ( $\exists s$ )( $P_1(s)$ )等价于  $\neg(\forall s)(\neg P_1(s))$ 。

(4)  $P_1 \rightarrow P_2$ 等价于  $\neg P_1 \vee P_2$ 。

关系代数表达式可以转换为元组表达式，例如， $R \cup S$ 可用 $\{t|R(t) \vee S(t)\}$ 表示， $R-S$ 可用 $\{t|R(t) \wedge \neg S(t)\}$ 表示。

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第5章：数据库系统基础知识

作者：希赛教育软考学院 来源：希赛网 2014年05月04日

## 一点一练

### 试题1

若对关系R (A, B, C, D) 进行 $\pi_{1,3}(R)$ 运算，则该关系运算与(1)等价，表示(2)。

(1) A.  $\pi_{A=1,C=3}(R)$  B.  $\pi_{A=1 \wedge C=3}(R)$

C.  $\pi_{A,C}(R)$  D.  $\pi_{A=1 \vee C=3}(R)$

(2) A. 属性A和C的值分别等于1和3的元组为结果集

B. 属性A和C的值分别等于1和3的两列为结果集

C. 对R关系进行A=1, C=3的投影运算

D. 对R关系进行属性A和C的投影运算

### 试题2

若关系R、S如图5-3所示，则R与S自然连接后的属性列数和元组个数分别为(3)；

$\pi_{1,4}(\sigma_{3=6}(R \times S))$  (4)。

关系 R				关系 S	
A	B	C	D	C	D
a	b	c	d	c	d
a	c	d	c	g	f
a	d	g	f		
a	b	g	f		

图5-3关系R与S

(3) A. 4和3 B. 4和6 C. 6和3 D. 6和6

(4) A.  $\pi_{A,D}(\sigma_{C=D}(R \times S))$  B.  $\pi_{A,R,D}(\sigma_{S.C=R.D}(R \times S))$

C.  $\pi_{A,R,D}(\sigma_{R.C=S.D}(R \times S))$  D.  $\pi_{R,A,R,D}(\sigma_{S.C=S.D}(R \times S))$

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第5章：数据库系统基础知识

作者：希赛教育软考学院 来源：希赛网 2014年05月04日

## 解析与答案

## 试题1分析

本题主要考查关系运算。

投影运算是从一个关系里面抽取指明的属性（列）组成一个新的关系，这种运算是一种对列进行操作的运算。本题中 $\pi_{1,3}(R)$ 的含义就是从关系R中，选取第1列（A）和第3列（C）组成一个新的关系模式，因此第1空选C，而2空选D。

## 试题1答案

(1) C (2) D

## 试题2分析

本题考查数据库的关系运算，是常考的知识点。

自然连接是一种等值连接但并不是每一个等值连接都是自然连接，若R与S无公共属性，则 $R \bowtie S = R \times S$ 。另外， $R \bowtie R$ 等于R本身。

图5-3中，R有4个属性，S有2个属性且与R中的相同，所以进行自然连接后，会有4列，记录会有3条。所以第(3)空要选择A答案。

又因为式子 $\pi_{1,4}(\sigma_{3=6}(R \times S))$ 是将R和S先进行笛卡儿积，然后再选取和投影。所以第(4)空正确的答案是C选项。

## 试题2答案

(3) A (4) C

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第5章：数据库系统基础知识

作者：希赛教育软考学院 来源：希赛网 2014年05月04日

## 规范化理论

不同的人对于相同的东西可以建立不同的模型，如何衡量模型建立的好坏？换而言之，按照什么原则建立模型？

这个原则就是规范化理论。

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第5章：数据库系统基础知识

作者：希赛教育软考学院 来源：希赛网 2014年05月04日

## 考点精讲

### 1. 规范化理论的作用

前面已经介绍了什么是规范化理论，那么在数据库设计中，规范化理论有什么作用，解决了什么问题呢？下面我们通过一个实例来分析这个问题。

设有一个关系模式R ( SNAME , CNAME , TNAME TADDRESS ) , 其属性分别表示学生姓名、选修的课程名、任课教师姓名和任课教师地址。仔细分析一下 , 就会发现这个模式存在下列存储异常的问题 :

( 1 ) 数据冗余 : 如果某门课程有100个学生选修 , 那么在R的关系中就要出现100个元组 , 这门课程的任课教师姓名和地址也随之重复出现100次。

( 2 ) 修改异常 : 由于上述冗余问题 , 当需要修改这个教师的地址时 , 就要修改100个元组中的地址值 , 否则就会出现地址值不一致的现象。

( 3 ) 插入异常 : 如果不知道听课学生名单 , 这个教师的任课情况和家庭地址就无法进入数据库 ; 否则就要在学生姓名处插入空值。

( 4 ) 删除异常 : 如果某门课程的任课教师要更改 , 那么原来任课教师的地址将随之丢失。

因此 , 关系模式R虽然只有四个属性 , 但却是性能很差的模式。产生这些异常的原因与关系模式属性值之间的联系直接有关。在模式R中 , 学生与课程有直接联系 , 教师与课程有直接联系 , 而教师与学生无直接联系 , 这就产生了模式R的存储异常。如果将R分解成下列两个关系模式 :

R1 ( SNAME , CNAME ) 和R2 ( CNAME , TNAME , TADDRESS ) , 则能消除上述的存储异常现象。

## 2 . 函数依赖

函数依赖是数据库的一种约束 , 决定了关系模式属于哪种范式。为了理解方面 , 下面我们先介绍函数依赖的有关概念。

设R(U)是属性U上的一个关系模式 , X和Y是U的子集 , r为R的任一关系 , 如果对于r中的任意两个元组u , v , 只要有u[X]=v[X] , 就有u[Y]=v[Y] , 则称X函数决定Y , 或称Y函数依赖于X , 记为  $X \rightarrow Y$ 。

例如 , 记录职工信息的结构如下 :

职工工号 ( EMP\_NO )

职工姓名 ( EMP\_NMAE )

所在部门 ( DEPT )。

我们说EMP\_NO函数决定EMP\_NMAE和DEPT , 或者说EMP\_NMAE , DEPT函数依赖于EMP\_NO , 记为 :  $EMP\_NO \rightarrow EMP\_NMAE$  ,  $EMP\_NO \rightarrow DEPT$ 。

## 3 . 键

超键 : 在关系模式中 , 能唯一标识元组的属性集称为超键。

候选键 : 如果一个属性集能唯一标识元组 , 且又不含有多余属性。

主键 : 关系模式中用户正在使用的候选键称为主键。

外键 : 如果公共关键字在一个关系中是主关键字 , 那么这个公共关键字被称为另一个关系的外键。

例如 , 记录职工信息的结构如下 :

职工工号 ( EMP\_NO )

职工身份证号 ( EMP\_CARDID )

职工姓名 ( EMP\_NMAE )

职工性别 ( EMP\_SEX )

所在部门编号 ( DEPT\_NO )。

在此关系中：

(1) EMP\_NO和EMP\_NMAE的组合是超键。

理由：EMP\_NO和EMP\_NMAE的组合能唯一标识元组的属性。

补充说明：其实只需要EMP\_NO就可以唯一标识元组的属性，所以EMP\_NO是本关系的候选键，也可以是本关系的主键。

(2) EMP\_NO 和EMP\_CARDID是候选键。

补充说明：一个关系的候选键可以有多个。

(3) EMP\_NO 或EMP\_CARDID可选作本关系的主键。

补充说明：一个关系的候选键有多个，但主键只能有一个。通常在候选键中选一个作为主键。

(4) DEPT\_NO为本关系相对于部门关系的外键。

理由：DEPT\_NO在本关系中不是主键，而在部门关系中是主键，所以它是本关系的外键。

补充说明：外键的作用是进行连接操作。例如现在要查找某个职工所在的部门名称，我们就需要用到外键来与部门关系进行连接，连接之后可以得到职工所在部门的名称。

#### 4. 图示法求候选键

求关系模式的候选键是进行范式界定的基础，也是系统分析员应该掌握的基本技能。使用候选键的定义“如果一个属性集能唯一标识元组，且又不含有多余属性。”来求解一个简单关系模式的候选键尚能应对，但面对复杂一些的关系模式，这种方法就不管用了。在此，引入一种求候选键的快捷方法，即图示法。

使用图示法求候选键，主要有三个步骤：

(1) 将关系的函数依赖关系，用“有向图”的方式表示。

(2) 找出入度为0的属性，并以该属性集合为起点，尝试遍历有向图，若能正常遍历图中所有结点，则该属性集即为关系模式的候选键。

(3) 若入度为0的属性集不能遍历图中所有结点，则需要尝试性的将一些中间结点（既有入度，也有出度的结点）并到入度为0的属性集中，直至该集合能遍历所有结点，集合为候选键。

例如，给定关系R(A1, A2, A3, A4)上的函数依赖集F={A1→A2, A3→A2, A2→A3, A2→A4}，现在要求R的候选关键字。

第一步，需要针对函数依赖集画出有向图，如图5-4所示。

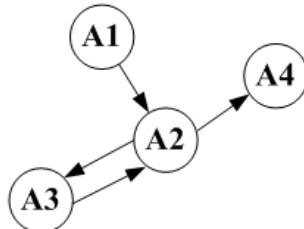


图5-4函数依赖集对应有向图

第二步，从该图中找出入度为0的结点，本图中入度为0的结点只有1个，即A1。通过尝试，可以发现从A1出发可以遍历有向图中的所有结点，所以本关系模式的候选键为A1。

#### 5. 范式

在数据库设计过程中，往往遇到数据冗余、修改异常、插入异常和删除异常等，为了设计一个好的数据库，人们定义了一些好的关系模式标准，称它们为规范的关系模式(简称范式，NF)。目前共定义了多个范式，分别为1NF、2NF、3NF、BCNF、4NF和5NF。但实际应用中，一般只要达到3NF。

如果有 $X \rightarrow U$ 在关系模式 $R(U)$ 上成立，并且不存在 $X$ 的任一真子集 $X'$ 使 $X' \rightarrow U$ 成立，那么称 $X$ 是 $R$ 的一个候选键。也就是 $X$ 值唯一决定关系中的元组。

在 $R(U)$ 中，如果 $X \rightarrow Y$ ，并且对于 $X$ 的任何一个真子集 $X'$ ，都有 $X' \rightarrow Y$ 不成立，则称 $Y$ 对 $X$ 完全函数依赖。若 $X \rightarrow Y$ ，但 $Y$ 不完全函数依赖于 $X$ ，则称 $Y$ 对 $X$ 部分函数依赖（注意：此处的 $X$ 是一个属性集）。

例如，如图5-5所示： $X_1 X_2 \rightarrow Y$ ，又有 $X_1 \rightarrow Y$ ，则 $Y$ 部分函数依赖于 $X_1 X_2$ 。

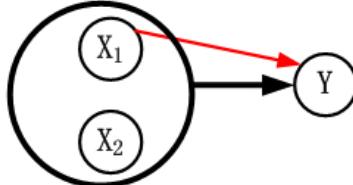


图5-5部分函数依赖示意图

在 $R(U)$ 中，如果 $X \rightarrow Y$ （ $Y$ 不是 $X$ 的真子集），且 $Y \rightarrow X$ 不成立， $Y \rightarrow Z$ ，则称 $Z$ 对 $X$ 传递函数依赖，如图5-6所示。

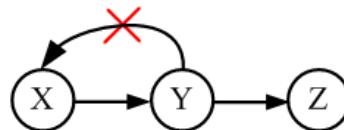


图5-6传递函数依赖示意图

有了上述的函数依赖概念之后，我们再介绍范式的概念。

第一范式（1NF）：在关系模式 $R$ 中，当且仅当所有域只包含原子值，即每个分量都是不可再分的数据项，则称实体 $E$ 是第一范式。本书中所引用的关系模式，一般都达到了第一范式的要求。

例如，图5-7所示的教师职称情况关系表就不满足1NF。

教师职称情况关系表		
系名称	高级职称人数	
	教授	副教授
计算机系	6	10
电子系	3	5

图5-7教师职称情况关系表

原因在于该关系模式中的“高级职称人数”不是一个原子属性，若将其拆分为“教授”和“副教授”两个属性，则满足1NF。满足1NF的关系模型会有许多重复值，并且增加了修改其数据时引起疏漏的可能性。为了消除这种数据冗余和避免更新数据的遗漏，我们需要更加规范的第二范式（2NF）。

第二范式（2NF）：当且仅当实体 $E$ 是第一范式（1NF），且每一个非键属性完全依赖主键（没有不完全依赖）时，则称实体 $E$ 是第二范式。

非键属性完全依赖又称部分函数依赖。

例如，有选课关系模式 $SC(Sno, Cno, Grade, Credit)$ ，其中： $(Sno, Cno) \rightarrow Grade$ ， $Cno \rightarrow Credit$ 。

由此可知 $SC$ 的主键为： $(Sno, Cno)$ 。这样 $(Sno, Cno) \rightarrow Credit$ 就被称为 $Credit$ 对主键 $(Sno, Cno)$ 的部分函数依赖，即 $Credit$ 属性只需要主键的一部分 $Cno$ 即可确定。所以本关系模式不符合2NF，只是1NF。若要将该关系模式转化为2NF，可以将关系模式拆分为： $SC1(Sno, Cno, Grade)$ 和 $SC2(Cno, Credit)$ 。

拆分前的SC与拆分后的SC1与SC2如图5-8所示，从该分解可以发现Credit在SC中有大量数据重复，而分解之后，解决了该问题。

SC 关系			
Sno	Cno	Grade	Credit
S01	C01	75	4
S02	C01	92	4
S03	C01	87	4
S04	C01	55	4
S01	C02	87	2
S02	C02	95	2
S01	C03	94	5
...	...	...	...

关系 SC1		
Sno	Cno	Grade
S01	C01	75
S02	C01	9
S03	C01	87
S04	C01	55
S01	C02	87
S02	C02	95
S01	C03	94
...	...	...

关系 SC2	
Cno	Credit
C01	4
C02	2
C03	5
...	...

图5-8关系SC及由此拆分的SC1与SC2

第三范式 ( 3NF ) : 当且仅当实体E是第二范式 ( 2NF ) , 且E中没有非主属性传递依赖于码时, 则称实体E是第三范式。

例如, 学生关系Student ( Sno , Sname , Dno , Dname , Location ) 各属性分别代表学号, 姓名, 所在系, 系名称, 系地址。其数据如表所示, 从关系模式中各属性之间的联系可以判断出本关系模式的函数依赖有: Sno→Sname , Sno→Dno , Sno→Dname , Sno→Location , Dno→Dname , Dno→Location。显然, Sno为主键。在函数依赖中有: Sno→Dno→Dname与 Sno→Dno→Location。这便是传递函数依赖。由于Dname与Location为非主属性, 同时传递依赖于码, 所以本关系模式不满足3NF。若要满足3NF, 需要将关系模式拆分为: Student ( Sno , Sname , Dno ) 和Department ( Dno , Dname , Location ) 。

关系 Student				
Sno	Sname	Dno	Dname	Location
S01	张三	D01	计算机系	1号楼
S02	李四	D01	计算机系	1号楼
S03	王五	D01	计算机系	1号楼
S04	赵六	D02	信息系	2号楼
...	...	...	...	...

图5-9关系Student

BCNF : 如果关系模型R∈1NF, 且R中每一个函数依赖关系中的决定因素都包含码, 则R是满足BC范式的关系, 记作R□BCNF。

按此定义, 消除了任何属性对码的传递依赖与部分依赖即为BCNF。但要判断任何属性是否对码存在传递依赖并不是一件容易的事情。

例如, 有关系模式P(C , S , T , R) , 根据语义有函数依赖F={C→T , ST→R , TR→C} , 现在需要

判断该关系模式是否满足BCNF。

此时我们需要分析其是否存在传递依赖。根据之前的图示法求候选键的方法，我们可以先画出相应的函数依赖图。

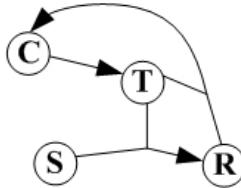


图5-10关系模式P函数依赖对应有向图

对该图进行分析，可以得知本关系模式的候选键有：(S, T) 和 (S, C)，所以主属性为：S、T、C，非主属性为：R。但此时由于属性之间的联系错综复杂，所以要界定关系模式是否存在传递函数依赖并不容易。为了准确的界定某关系模式是否为BCNF，我们需要引入另外的一些判别方法。

一个BCNF的关系模式，必须满足以下条件：

- 所有非主属性对每一个码都是完全函数依赖。
- 所有的主属性对每一个不包含它的码，也是完全函数依赖。
- 没有任何属性完全函数依赖于非码的任何一组属性，即每一个函数依赖的左部都必须包含候选键。

这也就是说，若某关系模式不满足以上任意一条，则它不满足BCNF。

由于有C→T，而C不包含候选键，所以本关系模式不满足BCNF。

根据各范式的定义，各级范式之间存在如下关系：

$$1NF \supset 2NF \supset 3NF \supset BCNF$$

## 6. 关系模式分解

如果某关系模式存在存储异常等问题，则可通过分解该关系模式来解决问题。将一个关系模式分解成几个子关系模式，需要考虑的是该分解是否保持函数依赖，是否是无损联接。

### (1) 无损连接分解

在理解无损连接分解前，我们需要明确一个概念：什么叫有损，什么又是无损？

答案很简单：有损——分解后不能还原；无损——分解后可以还原。

有了这个概念基础之后，理解无损连接分解就容易了。其定义如下：设R是一个关系模式，F是R上的一个函数依赖集。R分解成数据库模式 $\delta = \{R_1, \dots, R_k\}$ 。如果对R中每个满足F的关系r都有下式成立：

$$r = \pi_{R_1}(r) \bowtie \pi_{R_2}(r) \bowtie \dots \bowtie \pi_{R_k}(r)$$

则称分解 $\delta$ 相对于F是无损联接分解，否则称为损失联接分解。

要根据上述定义来判断一个分解是否是无损联接，这是一件很困难的事情，所以常常有两种方法用来解决这个问题。

### 方法一：公式法。

优点：速度快。

缺点：只能应用于一分为二的情况，一个关系模式要分解为三个关系模式就用不上了。

判定定理：设 $\rho = \{R_1, R_2\}$ 是R的一个分解，F是R上的函数依赖集，那么分解 $\rho$ 相对于F是无损联接分解的充要条件是 $(R_1 \cap R_2) \rightarrow (R_1 - R_2)$ 或 $(R_1 \cap R_2) \rightarrow (R_2 - R_1)$ 。要注意的是，这两个条件只要有任意一个条件成立就可以了。

## 方法二：表格法。

优点：速度慢。

缺点：适用于所有形式的模式分解。

对于表格法，由于其方法复杂，所以在此用一个例题进行讲解。

例：

设关系模式  $R(U, F)$ ，其中  $R$  上的属性集  $U = \{A, B, C, D, E\}$ ， $R$  上的函数依赖集

$F = \{A \rightarrow B, DE \rightarrow B, CB \rightarrow E, E \rightarrow A, B \rightarrow D\}$ 。分别判断： $p1 = \{R_1(AC), R_2(ED), R_3(AB)\}$ 、 $p2 = \{R_1(ABC), R_2(ED), R_3(ACE)\}$  是否为无损连接？

首先可以利用之前讲到的图示法求出本题的候选键，求得关系  $R$  有三个候选键：BC、EC，AC。

接下来是判断模式分解过程中的无损连接的问题。这个问题相对来说比较复杂。需要列表来进行分析。

对  $p1 = \{R_1(AC), R_2(ED), R_3(AB)\}$  构造初始的判定表如表5-4所示。

表5-4模式分解p1初始判定表

分解的关系模式	A	B	C	D	E
$R_1(AC)$	$a_1$	$b_{12}$	$a_3$	$b_{14}$	$b_{15}$
$R_2(ED)$	$b_{21}$	$b_{22}$	$b_{23}$	$a_4$	$a_5$
$R_3(AB)$	$a_1$	$a_2$	$b_{33}$	$b_{34}$	$b_{35}$

由于  $A \rightarrow B$ ，属性 A 的第 1 行和第 3 行相同，可以将第 1 行  $b_{12}$  改为  $a_2$ ；又由于  $B \rightarrow D$ ，属性 B 的第 1 行和第 3 行相同，所以需要将属性 D 第 1 行  $b_{14}$  和第 3 行  $b_{34}$ ，改为同一符号，即取行号值最小的  $b_{14}$ 。

修改后的判定表如表5-5所示。

表5-5模式分解p1修改判定表

分解的关系模式	A	B	C	D	E
$R_1(AC)$	$a_1$	$a_2$	$a_3$	$b_{14}$	$b_{15}$
$R_2(ED)$	$b_{21}$	$b_{22}$	$b_{23}$	$a_4$	$a_5$
$R_3(AB)$	$a_1$	$a_2$	$b_{33}$	$b_{14}$	$b_{35}$

反复检查函数依赖集  $F$ ，无法修改上表，所以是有损连接的。

对  $p2 = \{R_1(ABC), R_2(ED), R_3(ACE)\}$  构造初始的判定表如表5-6 所示。

表5-6模式分解p2初始判定表

分解的关系模式	A	B	C	D	E
$R_1(ABC)$	$a_1$	$a_2$	$a_3$	$b_{14}$	$b_{15}$
$R_2(ED)$	$b_{21}$	$b_{22}$	$b_{23}$	$a_4$	$a_5$
$R_3(ACE)$	$a_1$	$b_{32}$	$a_3$	$b_{34}$	$a_5$

由于  $A \rightarrow B$ ，属性 A 的第 1 行和第 3 行相同，可以将第 3 行  $b_{32}$  改为  $a_2$ ； $E \rightarrow A$ ，属性 E 的第 2 行和第 3 行相同，可以将属性 A 第 2 行  $b_{21}$  改为  $a_1$ ； $AC \rightarrow E$ ，属性 E 的第 2 行和第 3 行相同，可以将属性 E 第 1 行  $b_{15}$  改为  $a_5$ ； $B \rightarrow D$ ，属性 B 的第 1 行和第 3 行相同，所以需要将属性 D 第 1 行  $b_{14}$  和第 3 行  $b_{34}$ ，改为同一符号，即取行号值最小的  $b_{14}$ 。 $E \rightarrow D$ ，属性 E 的第 1 ~ 3 行相同，可以将属性 D 第 1 行  $b_{14}$  和第 3 行  $b_{34}$  改为  $a_4$ 。修改后的判定表如表5-7所示。

表5-7模式分解p2修改判定表

分解的关系模式	A	B	C	D	E
$R_1(ABC)$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$
$R_2(ED)$	$a_1$	$b_{22}$	$b_{23}$	$a_4$	$a_5$
$R_3(ACE)$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$

由于上表第一行全为a，故分解无损。

### (2) 保持函数依赖分解

设数据库模式 $\delta=\{R_1, \dots, R_k\}$ 是关系模式R的一个分解，F是R上的函数依赖集， $\delta$ 中每个模式 $R_i$ 上的函数依赖集是 $F_i$ 。如果 $\{F_1, F_2, \dots, F_k\}$ 与F是等价的（即相互逻辑蕴涵），则称分解 $\delta$ 保持函数依赖。如果分解不能保持函数依赖，则 $\delta$ 的实例上的值就可能有违反函数依赖的现象。

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第5章：数据库系统基础知识

作者：希赛教育软考学院 来源：希赛网 2014年05月04日

## 一点一练

### 试题1

给定关系模式 $R<U, F>$ ， $U = \{A, B, C\}$ ， $F = \{AB \rightarrow C, C \rightarrow B\}$ 。关系 $R_{(1)}$ ，且分别有 $_{(2)}$ 。

- (1) A. 只有1个候选关键字AC      B. 只有1个候选关键字AB
  - C. 有2个候选关键字AC和BC      D. 有2个候选关键字AC和AB
- (2) A. 1个非主属性和2个主属性B. 2个非主属性和1个主属性
  - C. 0个非主属性和3个主属性D. 3个非主属性和0个主属性

### 试题2

设有学生实体Students（学号，姓名，性别，年龄，家庭住址，家庭成员，关系，联系电话），其中“家庭住址”记录了邮编、省、市、街道信息；“家庭成员，关系，联系电话”分别记录了学生亲属的姓名、与学生的关系以及联系电话。

学生实体Students中的“家庭住址”是一个 $_{(3)}$ 属性；为使数据库模式设计更合理，对于关系模式Students $_{(4)}$ 。

- (3) A. 简单      B. 多值      C. 复合      D. 派生
- (4) A. 可以不作任何处理，因为该关系模式达到了3NF
- B. 只允许记录一个亲属的姓名、与学生的关系以及联系电话的信息
- C. 需要对关系模式Students增加若干组家庭成员、关系及联系电话字段
- D. 应该将家庭成员、关系及联系电话加上学生号，设计成为一个独立的实体

### 试题3

建立一个供应商、零件数据库。其中“供应商”表S(Sno,Sname,Zip,City)分别表示：供应商代码、供应商名、供应商邮编、供应商所在城市，其函数依赖为： $Sno \rightarrow (Sname, Zip, City)$ ， $Zip \rightarrow City$ 。“供应商”表S属于 $_{(5)}$ 。

- (5) A. 1NF B. 2NF C. 3NF D. BCNF

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

## 解析与答案

### 试题1分析

本题主要考查函数依赖的基础知识。

关系中的某一属性或属性组的值能唯一的标识一个元组，而其任何真子集都不能再标识，则称该属性组为候选码。

但这里大家要注意，如果一个关系有多个不同的主码时，那么这些主码组合在一起就是候选码，也就是说一个关系的主码只能选一个，而候选码可以有多个，这就好比选总统一样，候选人可以多个，但最终的总统只能有一个，当然也有些地方的候选人就只有一个，候选码也一样，也有可能只有一个，在这种情况下，候选码就是主码。

主属性和非主属性是互补的，一个关系模式中的属性不是主属性就是非主属性。组成候选码的属性就是主属性，其它的就是非主属性，所以要判断关系模式中的属性是主属性还是非主属性，首先要求解出其候选码。

在本题中，从题目给出的函数依赖关系我们可以看出，AB能推导出C，即能推导出所有的属性；而C能推导出B，同样，AC也能推导出所有的属性，因此AB与AC都是该关系的候选码。所以该关系中的属性都是主属性，没有非主属性。

### 试题1答案

(1) D (2) C

### 试题2分析

在本题中，由于题目告诉我们“家庭住址”记录了邮编、省、市、街道信息，那么家庭家庭住址属性可以再分，因此它是一个复合属性。派生属性是指通过其他属性可以计算获得结果的属性。

为使数据库模式设计更合理，我们可以将家庭成员、关系及联系电话加上学号，设计成为一个独立的实体，这里大家要注意一定要加上学号，因为学号唯一标识一名学生。

### 试题2答案

(3) C (4) D

### 试题3分析

这是一道考查数据库范式的题目。表S中的每一个属性都是不可再分的，所以该关系表是符合第一范式的；并且每一个非主属性完全函数依赖于码Sno，所以它又符合第二范式；那是不是达到第三范式呢？如果是，则每一个非主属性既不部分依赖于码也不传递依赖于码，而事实上，Zip→City，表S中存在传递依赖。所以表S最高属于2NF。

### 试题3答案

(5) B

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

结构化查询语言SQL是关系数据库的标准语言，它是集DDL（数据定义）、DML（数据操纵）和数据控制功能（授权、完整性规则和事务控制语句）于一体的数据库语言。

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第5章：数据库系统基础知识

作者：希赛教育软考学院 来源：希赛网 2014年05月04日

## 考点精讲

### 1. 数据定义语言

SQL的DDL主要是包括SQL模式、基本表及视图的定义和撤销的操作。包括以下几种，如表5-8所示。

表5-8 数据定义语言

操作对象	操作方式		
	创 建	删 除	修 改
表（基本表）	CREATE TABLE	DROP TABLE	ALTER TABLE
视图	CREATE VIEW	DROP VIEW	
索引	CREATE INDEX	DROP INDEX	

创建基本表的语法格式如下：

```
CREATE TABLE <表名>
  (<列名><数据类型>[列级完整性约束条件]
  [ , <列名><数据类型>[列级完整性约束条件]]...
  [ , <表级完整性约束条件>]) ;
```

修改基本表的语法格式如下：

```
ALTER TABLE <表名>
  [ADD <新列名><数据类型>[列级完整性约束条件]]
  [DROP <表级完整性约束条件>]
  [MODIFY <列名><数据类型>] ;
```

### 2. 数据操纵语言

数据操纵语言可以分为数据查询和数据更新（插入、删除、更新）两大类操作。其中由SELECT语句实现的数据查询是核心操作，也是历年考试中最重要的考点。其语法如下：

```
SELECT [ALL|DISTINCT] <目标列表达式>[ , <目标列表达式>]...
  FROM <表或视图名>[ , <表或视图名>]...
  [WHERE <条件表达式>]
  [GROUP BY <列名1> [HAVING <条件表达式>]]
  [ORDER BY <列名2> [ASC|DESC]] ;
```

下面，我们将针对以下几个示例关系模式进行阐述。

student (Sno,Sname,Ssex,Sage,Sdept)：学生（学号，姓名，性别，年龄、所在系）

course (Cno,Cname,Cpno,Ccredit) : 课程 (课程号, 课程名, 基础课号, 学分)

SC (Sno,Cno,Grade) : 学生选修表 (学号, 课程号, 成绩)

单表查询 : 即只涉及一个表的查询 (如表5-9所示)。

表5-9 单表查询

操作类别	操作子类	示例	说明及等价代数关系
对列操作	查询指定列	SELECT Sno,Sname FROM student;	$\pi_{Sno,Sname}(\text{student})$ 投影运算
	查询全部列	SELECT * FROM student;	就是表本身
对行操作	消除重复行	SELECT DISTINCT Sno FROM SC;	
	单条件	SELECT Sname FROM student WHERE Sage<20;	查询年龄<20 的学生 $\pi_{Sname}(\sigma_{Sage<'20'}(\text{student}))$
	确定集合	SELECT Sname FROM student WHERE Sdept IN('CS','MA'); 注: IN(属于), NOT IN(不属于)	查询属于计算机系(CS)及数学系(MA)的学生 $\pi_{Sname}(\sigma_{Sdept='CS' \vee Sdept='MA'}(\text{student}))$
	多条件	SELECT Sname FROM student WHERE Sdept='CS' Sage<20;	$\pi_{Sname}(\sigma_{Sdept='CS' \wedge Sage<'20'}(\text{student}))$

除此之外, 我们还可以对查询结果进行以下处理 (如表5-10所示)。

表5-10 对查询结果进行处理

处理类型	处理子类	示例/语法
对结果排序	降序	加上: ORDER BY 排序字段名 DESC;
	升序	默认值, 关键字是 ASC
使用集函数 ( 放置 于 SELECT 子句 中)	统计记录总数	COUNT([DISTINCT ALL] *)
	统计一列中值的个数	COUNT([DISTINCT ALL] <列名>)
	计算一列中值的总和	SUM([DISTINCT ALL] <列名>)
	计算一列值的平均值	AVG([DISTINCT ALL] <列名>)
	求一列值中的最大值	MAX([DISTINCT ALL] <列名>)
	求一列值中的最小值	MIN([DISTINCT ALL] <列名>)
对结果分组	将查询结果按某一列或多列值分组	GROUP BY <列名>, 例如, 按课程号统计学生数, 即得到每个课程号的选修人数: SELECT Cno,Count(Sno)
对结果分组	将查询结果按某一列或多列值分组	FROM SC GROUP BY Cno; 还可以使用 HAVING <条件表达式>筛选, 如: HAVING COUNT(*)>5

联接查询 : 即同时涉及两个以上的表的查询。

它是关系数据库中最主要的查询, 它包括等值联接、自然联接、非等值联接、自身联接、外联接和复合条件联接查询等几大类, 如表5-11所示。

表5-11 联接查询

操作类别	示例	说明等价代数关系
等值联接 (注: 当联接运算符不是=就是非等值联接)	SELECT student.* ,SC.* FROM student,SC WHERE student.Sno=SC.Sno;	查询每个学生及其选修课程情况 $\sigma_{student.Sno=SC.Sno}(\text{student} \times \text{SC})$ 使用广义笛卡积运算
	SELECT student.Sno,Sname,Cno FROM student,SC WHERE student.Sno=SC.Sno;	查询同上 $\pi_{student.Sno,Sname,Cno}(\text{student} \times \text{SC})$ 使用自然联接运算
自身联接	SELECT A.Cno,B.Cpno FROM course A,course B WHERE A.Cpno=B.Cno;	查询每一门课程的基础选修课的基础选修课。

	(注: course A, course B 是为 course 定义了个别名, 分别为 A 和 B)	$\pi_{Cno, P_{con}}(\sigma_{course \geq \text{course}})$ 或 $\pi_{Cno, P_{con}}(\sigma_{1 \geq 1}(course \times course))$
外联接	<pre>SELECT student.Sno, Sname, Cno FROM student, SC WHERE student.Sno=SC.Sno(*);</pre>	查询每个学生及其选修课程情况, 而在联接谓词中加上“*”号, 就将使用外联接, 即该查询会得出没有选课的学生信息。这个谓词在条件表达式的左边就是左外联接, 在右边就是右外联接 $\pi_{student, Sno, Sname, Cno}(student \setminus [SC])$
复合条件联接 (即在 WHERE 子句中有多个条件)	<pre>SELECT student.Sno, Sname FROM student, SC WHERE student.Sno=SC.Sno AND SC.Cno='2' AND SC.Grade&gt;90;</pre>	查询选修 2 号课程且成绩在 90 分以上的学 生。 $\pi_{Sno, Sname, Grade}(\sigma_{SC.Cno=2 \wedge SC.Grade>90}(student \times SC))$
	<pre>SELECT student.Sno, Sname, Grade FROM student, SC, course WHERE student.Sno=SC.Sno AND SC.Cno=Course.Cno AND Cname='数据库'; </pre>	查询选修课程“数据库”的学生信息及其成 绩。 $\pi_{Sno, Sname, Grade}(\sigma_{Cname='数据库'}(student \times SC \times course))$

从上表中, 我们可以发现在SQL与关系代数表达式的转换中, 有这样几个规律: SELECT后面的<列名>一般体现为投影运算; FROM后面列出了关系模式, WHERE子句的等值部分通常是转为“自然联接”、“笛卡尔积”等运算;而非等值比较则通常是转为“选择”运算。

集合查询: SELECT语句的查询结果是元组(记录)的集合, 所以多个SELECT语句的结果可以进行集合操作。SQL语言提供的集合操作有并操作UNION、交操作INTERSECT和差操作MINUS。

数据更新操作(如表5-12所示)

表5-12 数据更新操作

操作类别	语法	示例
插入数据	<pre>INSERT INTO &lt;表名&gt; [&lt;属性列 1&gt; [, &lt;属性列 2&gt;...]] VALUES(&lt;常量 1&gt; [, &lt;常量 2&gt;] ...); 注: VALUES 子句可用子查询替代</pre>	<pre>INSERT INTO student VALUES('95020', '李四', '男', 'CS', '18'); INSERT INTO SC(Sno, Cno) VALUES('95020', '1');</pre>
修改数据	<pre>UPDATE &lt;表名&gt; SET &lt;列名&gt;=&lt;表达式&gt; [, &lt;列名&gt;=&lt;表达式&gt;] ... [WHERE &lt;条件&gt;];</pre>	将学生 95001 的年龄改为 23 岁 <pre>UPDATE student SET Sage=23 WHERE Sno='95001';</pre>
删除数据	<pre>DELETE FROM &lt;表名&gt; [WHERE &lt;条件&gt;];</pre>	删除学号为 95019 的学生记录 <pre>DELETE FROM student WHERE Sno='95019';</pre>

### 3. 权限赋予

GRANT { ALL [PRIVILEGES] }

| 权限 [(列名 [, ..., n])] [, ..., n]

[ ON [类名 :: ] 安全对象 ] TO 安全主体 [, ..., n]

[ WITH GRANT OPTION ] [ AS 安全主体 ]

例如:

GRANT CREATE DATABASE, CREATE TABLE TO Mary, John

本语句的作用是给用户Mary和John 赋予创建数据库与创建表的权限。

## 一点一练

### 试题1

若有关系R ( A , B , C , D ) 和S ( C , D , E ) , 则与表达式 $\pi_{3,4,7}(\sigma_{4<5}(R \times S))$ 等价的SQL语句如下：

- SELECT \_\_1\_\_FROM \_\_2\_\_WHERE \_\_3\_\_ ;
- ( 1 ) A . A,B,C,D,E      B . C,D,E  
 C . R.A, R.B,R.C, R.D, S.E      D . R.C,R.D,S.E
- ( 2 ) A . R      B . S      C . R , S      D . RS
- ( 3 ) A . D<C      B . R.D<S.C      C . R.D<R.C      D . S.D< R.C

### 试题2

将Students表的插入权限赋予用户UserA , 并允许其将该权限授予他人 , 应使用的SQL语句为 :

- GRANT \_\_4\_\_ TABLE Students TO UserA \_\_5\_\_;
- ( 4 ) A . UPDATE      B . UPDATEON      C . INSERT      D . INSERT ON  
 ( 5 ) A . FORALL      B . PUBLIC  
 C . WITH CHECK OPTION      D . WITH GRANT OPTION

### 试题3

某销售公司数据库的零件关系P ( 零件号 , 零件名称 , 供应商 , 供应商所在地 , 库存量 ) , 函数依赖集F={零件号→零件名称 , ( 零件号 , 供应商 )→库存量 , 供应商→供应商所在地 ) 。零件关系模式P属于\_\_6\_\_。

查询各种零件的平均库存量、最多库存量与最少库存量之间差值的SQL语句如下 :

- SELECT零件号 , 零件名称 , \_\_7\_\_ ,  
 FROM P  
 \_\_8\_\_ ;
- ( 6 ) A . 1NF      B . 2NF      C . 3NF      D . 4NF
- ( 7 ) A . AVG ( 库存量 ) AS平均库存量 , MAX ( 库存量 ) -MIN ( 库存量 ) AS差值  
 B . 平均库存量AS AVG ( 库存量 ) , 差值AS MAX ( 库存量 ) -MIN ( 库存量 )  
 C . AVG库存量AS平均库存量 , MAX库存量-MIN库存量AS差值  
 D . 平均库存量AS AVG库存量 , 差值AS MAX库存量-MIN库存量
- ( 8 ) A . ORDER BY供应商 B . ORDER BY零件号  
 C . GROUP BY供应商 D . GROUP BY零件号

## 解析与答案

### 试题1分析

$R \times S$  表示关系R与关系S的笛卡尔积，其结果是一个具有n ( n的值由关系R与S的元组个数共同决定 ) 行7列元素的二维表， $\sigma_{4 < 5} ( R \times S )$  是选取表中第4列 ( R.D ) 元素值小于第5列 ( S.C ) 元素值的记录，然后对这个结果进行投影操作，即选取第3 ( R.C ) 、4 ( R.D ) 、7 ( S.E ) 列组成一个新表。

因此第1空的结果应该为R.C , R.D , S.E，而第2空是从关系表R和S中进行查找，所以应该为R , S。查找的条件为R.D元素值小于S.C元素值，即R.D < S.C。

### 试题1答案

( 1 ) D ( 2 ) C ( 3 ) B

### 试题2分析

赋予用户权限命令的简单格式可以概括为：GRANT 权限ON 数据库对象TO 用户 WITH GRANT OPTION。而WITH GRANT OPTION是可选项，表示用户可以将其权限通过GRANT语句授权给其它用户。因此本题的正确答案应分别是INSERT ON和WITH GRANT OPTION。

### 试题2答案

( 4 ) D ( 5 ) D

### 试题3分析

要求一个关系模式属于第几范式，一个很重要的步骤就是求出该关系模式的主键，在本题中，根据函数依赖集F，我们不难知道，零件关系P的主键为（零件号，供应商），因为从这两个属性出发可以推导出所有其它属性。在求出主键后，我们再看是否存在部分依赖，很显然零件号—零件名称是部分依赖，因此该关系模式不满足2NF，而只满足1NF。

对于第 ( 7 ) 空，根据题目要求，是要显示各种零件的平均库存量以及最多库存量与最少库存量之间差值，求平均库存量可以用函数AVG，而求最多库存量和最小库存量分别用函数MAX和MIN。AS是取别名的意思，因此本题答案选A。

根据题目意思，是要显示各零件的平均库存量以及最多库存量与最少库存量之间差值，那么这里就需要按零件的种类对查询的结果进行分组，因此第 ( 8 ) 空应该填GROUP BY零件号。

### 试题3答案

( 6 ) A ( 7 ) A ( 8 ) D

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

### 试题1

从数据库管理系统的角度看，数据库系统一般采用如图5-11所示的三级模式结构。图中①②处应填写\_\_(1)\_，③处应填写\_\_(2)\_。

- (1) A. 外模式/概念模式 B. 概念模式/内模式  
C. 外模式/概念模式映像 D. 概念模式/内模式映像
- (2) A. 外模式/概念模式 B. 概念模式/内模式  
C. 外模式/概念模式映像 D. 概念模式/内模式映像

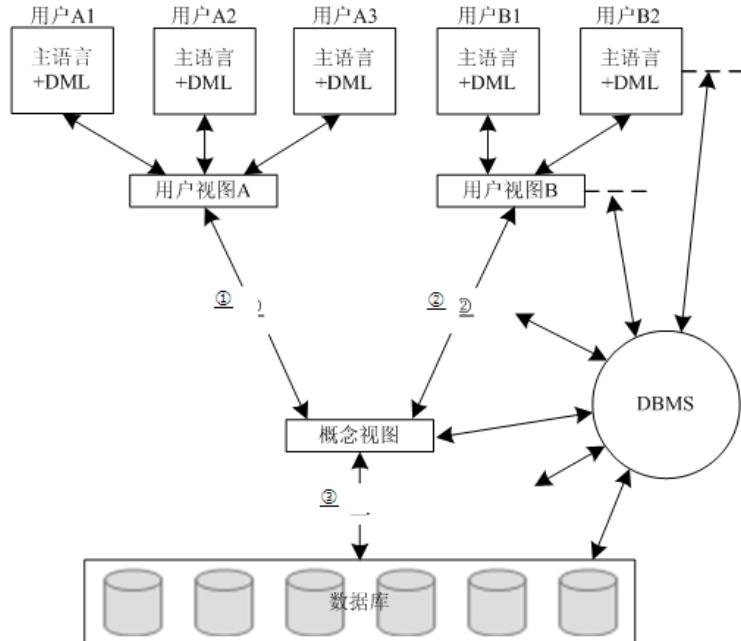


图5-11数据库三级模式结构

### 试题2

某医院数据库的部分关系模式为：科室（科室号，科室名，负责人，电话）、病患（病历号，姓名，住址，联系电话）和职工（职工号，职工姓名，科室号，住址，联系电话）。假设每个科室有一位负责人和一部电话，每个科室有若干名职工，一名职工只属于一个科室；一个医生可以为多个病患看病；一个病患可以由多个医生多次诊治。科室与职工的所属联系类型为\_\_(3)\_，病患与医生的就诊联系类型为\_\_(4)\_。对于就诊联系最合理的设计是\_\_(5)\_，就诊关系的主键是\_\_(6)\_。

- (3) A. 1:1 B. 1:n C. n:1 D. n:m
- (4) A. 1:1 B. 1:n C. n:1 D. n:m
- (5) A. 就诊（病历号，职工号，就诊情况）  
B. 就诊（病历号，职工姓名，就诊情况）  
C. 就诊（病历号，职工号，就诊时间，就诊情况）  
D. 就诊（病历号，职工姓名，就诊时间，就诊情况）
- (6) A. 病历号，职工号 B. 病历号，职工号，就诊时间  
C. 病历号，职工姓名 D. 病历号，职工姓名，就诊时间

### 试题3

设有关系模式R(课程, 教师, 学生, 成绩, 时间, 教室)，其中函数依赖集F如下：

- $F = \{ \text{课程} \rightarrow \rightarrow \text{教师}, (\text{学生}, \text{课程}) \rightarrow \text{成绩}, (\text{时间}, \text{教室}) \rightarrow \text{课程}, (\text{时间}, \text{教师}) \rightarrow \text{教室}, (\text{时间}, \text{学生}) \rightarrow \text{教室} \}$

关系模式R的一个主键是\_\_(7)\_，R规范化程度最高达到\_\_(8)\_。若将关系模式R分解为3个关系模式R1(课程, 教师)、R2(学生, 课程, 成绩)、R3(学生, 时间, 教室, 课程)，其中R2的规范化程度最高达到\_\_(9)\_。

(7) A. (学生, 课程) B. (时间, 教室) C. (时间, 教师) D. (时间, 学生)

(8) A. 1NF B. 2NF C. 3NF D. BCNF

(9) A. 2NF B. 3NF C. BCNF D. 4NF

#### 试题4

假设有学生S(学号, 姓名, 性别, 入学时间, 联系方式), 院系D(院系号, 院系名称, 电话号码, 负责人)和课程C(课程号, 课程名)三个实体, 若一名学生属于一个院系, 一个院系有多名学生; 一名学生可以选择多门课程, 一门课程可被多名学生选择, 则图5-13中(a)和(b)分别为\_\_(10)\_联系。假设一对多联系不转换为一个独立的关系模式, 那么生成的关系模式\_\_(11)\_。

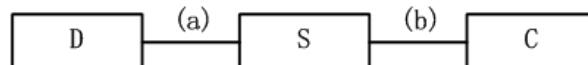


图5-13 ER图

(10) A. 1 \* 和 1 \* B. 1 \* 和 \* 1 C. 1 \* 和 \* 1 D. \* 1 和 \* \*

(11) A. S中应加入关系模式D的主键 B. S中应加入关系模式C的主键

C. D中应加入关系模式S的主键 D. C中应加入关系模式S的主键

#### 试题5

设关系模式R(A, B, C), 传递依赖指的是\_\_(12)\_; 下列结论错误的是\_\_(13)\_。

(12) A. 若  $A \rightarrow B$ ,  $B \rightarrow C$ , 则  $A \rightarrow C$  B. 若  $A \rightarrow B$ ,  $A \rightarrow C$ , 则  $A \rightarrow BC$

C. 若  $A \rightarrow C$ , 则  $AB \rightarrow C$  D. 若  $A \rightarrow BC$ , 则  $A \rightarrow B$ ,  $A \rightarrow C$

(13) A. 若  $A \rightarrow BC$ , 则  $A \rightarrow B$ ,  $A \rightarrow C$  B. 若  $A \rightarrow B$ ,  $A \rightarrow C$ , 则  $A \rightarrow BC$

C. 若  $A \rightarrow C$ , 则  $AB \rightarrow C$  D. 若  $AB \rightarrow C$ , 则  $A \rightarrow C$ ,  $B \rightarrow C$

#### 试题6

希赛公司学生、教师和课程实体对应的关系模式如下:

学生(学号, 姓名, 性别, 年龄, 家庭住址, 电话)

课程(课程号, 课程名)

又由于教师和课程之间是一个\_\_(16)\_的联系, 所以\_\_(17)\_。

(14) A. 1对1 B. 1对多 C. 多对1 D. 多对多

(15) A. 不需要增加一个新的关系模式

B. 不需要增加一个新的关系模式, 只需要将1端的码插入多端

C. 需要增加一个新的选课关系模式, 该模式的主键应该为课程号

D. 需要增加一个新的选课关系模式, 该模式的主键应该为课程号和学号

(16) A. 1对1 B. 1对多 C. 多对1 D. 多对多

(17) A. 不需要增加一个新的关系模式, 只需要将职工号插入课程

B. 不需要增加一个新的关系模式, 只需要将课程号插入教师关系模式

- C . 需要增加一个新的选课关系模式 , 该模式的主键应该为课程号
- D . 需要增加一个新的选课关系模式 , 该模式的主键应该为课程号和学号

版权方授权希赛网发布 , 侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第 5 章 : 数据库系统基础知识

作者 : 希赛教育软考学院 来源 : 希赛网 2014 年 05 月 05 日

## 习题解析

### 试题1分析

本题考查数据库系统的模式 , 是常考的知识点。考的是基本的概念 , 具体概念参看本章 “5.2 数据库模式及 ER 模型” 。

### 试题1答案

( 1 ) C ( 2 ) D

### 试题2分析

本题主要考查关系模式的基础知识。

在本题中 , 题目告诉我们每个科室有一位负责人和若干名职工 , 而一名职工只属于一个科室 , 那么很容易我们就能知道科室与职工的所属联系类型为 1:n 。

另外 , 题目告诉我们一个医生可以为多个病患看病 , 一个病患可以由多个医生多次诊治 , 所以病患与医生的就诊联系类型为多对多。

根据题目意思 , 就诊应该是病患与医生之间的联系 , 他们之间的联系是多对多 , 因此其联系要转换为独立的关系模式时 , 应该包含病患和医生关系模式的主键及自身的一些属性 , 如就诊时间 , 就诊情况。而病患的主键是病历号 , 而职工关系模式的主键为职工号 , 因此就诊关系模式为就诊 ( 病历号 , 职工号 , 就诊时间 , 就诊情况 ) , 而该关系模式的主键是 ( 病历号 , 职工号 , 就诊时间 ) , 因为这样才能唯一标识一条记录。至于主键为什么不是 ( 病历号 , 职工号 ) , 是因为存在同一个病人多次看同一个医生的情况 , 所以 ( 病历号 , 职工号 ) 不能唯一标识一条记录。

### 试题2答案

( 3 ) B ( 4 ) D ( 5 ) C ( 6 ) B

### 试题3分析

通过主键应该是可以推导出关系中的所有其它属性的 , 那么通过函数依赖集 F 我们可以知道 , 通过 ( 时间 , 学生 ) 可以推导出教室 , 然后通过 ( 时间 , 教室 ) 可以推导出课程 , 通过 ( 学生 , 课程 ) 又能推导出成绩 , 通过课程又可能推导出教师 , 所有其它属性都可以被推导出来 , 因此 ( 时间 , 学生 ) 是主键。

从函数依赖集 F 我们可以看到 , 没有对主键中单个属性的依赖关系 , 所有不存在非主属性对码的部分依赖 , 但存在传递依赖 , 因此关系模式只满足第 2 范式。

分解后的 R2 只包含了学生 , 课程及成绩三个属性 , 它们之间的依赖关系是 ( 学生 , 课程 ) → 成绩 , 那么主键应该是 ( 学生 , 课程 ) , 这里不存在部分依赖 , 也不存在传递依赖 , 因此满足 BC 范式。

### 试题3答案

( 7 ) D ( 8 ) B ( 9 ) C

#### 试题4分析

本题考查的是应试者对数据库系统中E-R模型方面的基础知识。

根据题意，学生S关系中一个学生属于一个院系，一个院系有多个学生，因此学生S和部门院系D之间是一个一对多的联系，故(a)应为1 \*；又因为一个学生可以选择多门课程，一门课程可由多名学生选择，所以学生S和课程C关系之间是一个多对多的联系，故(b)应为\* \*。

假设一对多联系(a)不转换为一个独立的关系模式，那么按照E-R模型向关系模型转换的规则，应当将一端的主键并入多端。所以，生成的关系模式学生S中应加入关系模式D的主键。

#### 试题4答案

( 10 ) C ( 11 ) A

#### 试题5分析

第(12)空，传递依赖：如果X->Y，Y->A，且Y不依赖X和A不是Y的子集，那么称X->A是传递依赖。A答案正确；B答案满足函数依赖的合并规则；C答案满足函数依赖的增广律；D答案满足引理规则。第(13)空，很明显是D答案错误，因为AB依赖于C，不可能得到A、B都完全依赖于C。

#### 试题5答案

( 12 ) A ( 13 ) D

#### 试题6分析

该题考查实体之间的基本关系，以及实体联系模型（ER模型）向关系模式的转换。实体之间的基本关系包括：1:1、1:n、m:n，而ER模型向关系模式转换的规则如下：

有了以上的理论基础，解题也就非常容易了。由于“一个学生可以选修多门课程，一门课程可以有多个学生选修”，所以学生与课程之间是多对多的联系。由于是多对多的联系，所以必须新增一个新的关系模式，用于记录联系的数据，这个关系模式中应包含两端关系模式的主键及联系自身的属性，主键为两端主键的组合。由于“一个教师只能讲授一门课程，但一门课程可以有多个教师讲授”，所以教师与课程的关系是多对一。多对一的情况可以不新建关系模式，而将联系的属性及1端的主键加入到n端。

#### 试题6答案

( 14 ) D ( 15 ) D ( 16 ) C ( 17 ) A

版权方授权希赛网发布，侵权必究

[上一节](#) [本书简介](#) [下一节](#)

第6章：网络基础知识

作者：希赛教育软考学院 来源：希赛网 2014年05月05日

### 考点脉络

根据考试大纲，本章要求考生掌握以下几个方面的知识点。

- ( 1 ) 网络体系结构。
- ( 2 ) 传输介质，传输技术，传输方法，传输控制。
- ( 3 ) 常用网络设备和各类通信设备的特点。
- ( 4 ) LAN（拓扑，存取控制，组网，网间互连）。