

## 习 题

### 一、填空题

1. 数据管理技术经历了人工管理，文件管理和数据库管理三个阶段。
2. 数据库是长期存储在计算机内、有组织的、统一管理的、可共享的相关数据的集合。
3. 数据库系统一般由数据库、数据库管理系统、应用系统、数据库管理员等构成。
4. 在数据库系统中，数据由 数据库管理系统 统一管理和控制。
5. 在数据库的三级模型结构中，对单个用户使用的数据视图的描述，称为外模式；对所有用户的公共数据视图的描述，称为模式；对物理存储数据视图的描述，称为内模式。
6. 数据库的 三级模型结构 和两级映像有力的保证了数据独立性的实现。
7. 数据的独立性包括物理独立性和逻辑独立性。
8. 外模式/模式间的映射提供了数据的逻辑独立性，模式/内模式映射提供了数据的物理独立性。
9. 数据库管理系统通常提供授权功能来控制不同用户访问数据的权限，这主要是为了实现数据库的安全性。
10. 从用户的角度来看，可以将数据库系统看作是由客户端和服务器两部分组成。

### 二、选择题

1. 数据库系统的特点是数据共享、数据独立、减少数据冗余、A和加强了数据保护。  
A. 避免数据不一致      B. 数据存储      C. 数据应用      D. 数据保密
2. 数据库的特点之一是数据的共享，严格地讲，这里的数据共享是指D。  
A. 同一个应用中的多个程序共享一个数据集合  
B. 多个用户、同一种语言共享数据  
C. 多个用户共享一个数据文件  
D. 多种应用、多种语言、多个用户相互覆盖地使用数据集合
3. 数据库系统的组成核心是B。  
A. 数据库      B. 数据库管理系统      C. 数据模型      D. 软件工具

4. A 是位于用户与操作系统之间的一层数据管理软件。
- A. 数据库管理系统 B. 数据库系统 C. 数据库 D. 数据库应用系统
5. 下列关于数据库系统的叙述中正确的是 A。
- A. 数据库系统减少了数据冗余
- B. 数据库系统避免了一切冗余
- C. 数据库系统中数据的一致性是指数据类型一致
- D. 数据库系统比文件系统能管理更多的数据
6. 数据库系统与文件系统的主要区别是 B。
- A. 数据库系统复杂，而文件系统简单
- B. 文件系统不能解决数据冗余和数据独立性问题，而数据库系统可以解决
- C. 文件系统只能管理程序文件，而数据库系统能够管理各种类型的文件
- D. 文件系统管理的数据量较少，而数据库系统可以管理庞大的数据量
7. 数据库系统的三级模式结构中，定义视图的组织方式属于 B。
- A. 概念模式 B. 外模式 C. 逻辑模式 D. 内模式
8. 数据库系统实现了数据独立性是因为采用了 B。
- A. 层次模型 B. 网状模型 C. 关系模型 D. 三级模式结构
9. 数据库三级模式体系结构的划分，有利于保持数据库的 A。
- A. 数据独立性 B. 数据安全性 C. 结构规范化 D. 操作可行性
10. 数据独立性是指 B。
- A. 数据之间相互独立
- B. 应用程序与 DB 的结构之间相互独立
- C. 数据的逻辑结构与物理结构相互独立
- D. 数据与磁盘之间相互独立
11. 物理独立性是指修改 B。
- A. 外模式，保持模式不变 B. 内模式，保持模式不变
- C. 模式，保持外模式不变 D. 模式，保持内模式不变
12. 在 DBS 中，DBMS 和 OS 之间的关系是 B。
- A. 相互调用 B. DBMS 调用 OS C. OS 调用 DBMS D. 并发运行

### 三、简答题

1. 利用计算机系统进行数据管理经历了哪三个阶段？各阶段的特点如何？

2. 利用数据库技术进行数据管理具有哪些优点？
3. 什么是数据库系统的数据独立性？包括哪两个方面？
4. 数据库系统通常有哪些部分组成？
5. 数据库管理系统具有哪些主要功能？
6. 数据库管理员（DBA）通常应具备的职责有哪些？
7. 数据库系统从逻辑上分为哪三级结构？每层所对应的模式结构描述的内容是什么？
8. 简述 DBMS 的工作模式和模块组成。

## 习 题

### 一、填空题

1. 数据模型包含三个要素：数据结构、数据操作和完整性约束。
2. 对现实世界进行第一层抽象的模型，称为概念模型；对现实世界进行第二层抽象的模型，称为数据模型。
3. 我们通常把层次模型和网状模型称为非关系模型。
4. 用树型结构表示实体类型及实体间联系的数据模型称为层次模型。
5. 概念模型属于信息世界的模型，实际上是现实世界到机器世界的一个中间层次。
6. 适合数据库集成的数据模型是半结构化数据模型。
7. 在面向对象数据模型中，通过消息实现对象之间的通信。
8. 在 XML 中，元素可以包括子元素和属性。

### 二、选择题

1. 在数据库技术中，独立于计算机系统的模型是A。  
A. E-R 模型      B. 层次模型      C. 关系模型      D. 面向对象的模型
2. 层次模型、网状模型和关系模型是根据 A 来划分的（命名）。  
A. 数据结构      B. 数据查询操作      C. 完整性约束      D. 数据更新操作
3. D不是面向对象数据模型中涉及到的概念。  
A. 类      B. 属性      C. 消息      D. 关系
4. 关于 XML 以下说法 C 正确的？  
A. 元素之间不能嵌套  
B. 同一元素中允许存在同名属性  
C. 可用 DTD 描述 XML 数据的结构  
D. 描述同类对象的数据结构必须相同

### 三、简答题

1. 试述数据模型的概念、数据模型的作用和数据模型的三要素。
2. 试述概念模型的作用。
3. 定义并解释概念模型中的以下术语：  
实体、实体型、实体集、属性、码
4. 试举出三个实例，要求两个实体集之间分别具有 1:1、1:n、m:n 的关系。

## 习 题

### 一、填空题

1. 关系数据模型中的关系可用二维表来表示,表中的一行对应关系的一个 元组,表中的一列对应关系的一个 属性。
2. 在关系代数运算中,使用 选择 运算可从关系中得到满足条件的元组;如果只对关系中的某些属性感兴趣,则可用关系代数的 投影 运算选择这些属性。
3. 设关系 R 和 S 分别有 m 和 n 个元组,  $k_1$  和  $k_2$  个属性,有  $k_3$  个相同的属性,则  $R \times S$  的元组个数是  $m \times n$ ,属性的个数是  $k_1 + k_2$ ;  $R \bowtie S$  的属性个数是  $K_1 + K_2 - K_3$ 。
4. 在关系 A (S, SN, D) 和 B (D, CN, NM) 中, S 是 A 的主键, A 中的属性 D 与 B 中的主键 D 相对应,则 D 在 A 中称为 外键。
5. 关系数据模型的实体完整性规则要求,关系的主属性 不能为空。
6. 有关系: 教学(学号, 教工号, 课程号)。假定每个学生可以选修多门课程,每门课程可以由多名学生来选修,每个老师只能讲授一门课程,每门课程可以由多个老师来讲授,那么该关系的主键是 (学号, 教工号) 或 (学号, 课程号)。
7. 有如下关系: 学生(学号, 姓名, 性别, 专业号, 年龄),将属性年龄的取值范围定义在 18~30 之间为 用户定义 完整性约束。
8. 关系模型中可以有三类完整性约束: 其中 实体 完整性和 参照 完整性是关系模型必须满足的完整性约束条件,应该由 RDBMS 自动支持。
9. 关系代数运算中的 5 种基本的操作包括: 并、差、笛卡尔积、投影和选择。
10. 关系操作的特点是 集合 操作。

### 二、选择题

1. 在关系代数的专门关系运算中,从表中取出满足条件的列的操作称为 B。  
A. 选择                  B. 投影                  C. 连接                  D. 扫描
2. 进行自然连接运算的两个关系必须具有 B。  
A. 相同属性个数                  B. 公共属性  
C. 相同关系名                  D. 相同关键字
3. 关系演算是用 A 来表达查询要求的方式。  
A. 谓词                  B. 关系的运算                  C. 元组                  D. 域
4. 在关系代数中,对一个关系做投影操作后,新关系的元组个数 B 原来关系的元组个数。  
A. 小于                  B. 小于或等于                  C. 等于                  D. 大于
5. 在关系数据库中,表与表之间的联系是通过定义 B 实现的。

- A. 实体完整性 B. 参照完整性
- C. 用户自定义的完整性 D. 值域
6. 关系模型中，一个候选码 C。
- A. 可由多个任意属性组成
- B. 至少由一个属性组成
- C. 可由一个或多个其值能惟一标识该关系模式中任何元组的属性组成
- D. 必须由多个属性组成
7. 集合 R 与 S 的交可以用关系代数的基本运算表示为 A。
- A.  $R - (R - S)$  B.  $R + (R - S)$
- C.  $R - (S - R)$  D.  $S - (R - S)$
8. 下列关系代数的操作中，不是基本运算的是 A。
- A. 交 B. 并
- C. 笛卡尔积 D. 投影
9. 以下关于关系性质的说法中，错误的是 B。
- A. 关系中任意两行的值不能完全相同 B. 关系中任意两列的值不能完全相同
- C. 关系中任意两行可以交换顺序 D. 关系中任意两列可以交换顺序
10. 在关系数据库中，实现表中任意两行不能相同的约束是依据 C。
- A. 外码 B. 属性 C. 主键 D. 列
11. 以下关于外键和相应的主键之间的关系，正确的是 A。
- A. 外键并不一定要与相应的主键同名
- B. 外键一定要与相应的主键同名
- C. 外键一定要与相应的主键同名而且惟一
- D. 外键一定要与相应的主键同名，但并不一定惟一
12. 关系中的主关键字不允许取空值是符合 A 约束规则。
- A. 实体完整性 B. 参照完整性 C. 用户定义的完整性 D. 数据完整性
13. 如果关系 R 中有 4 个属性和 3 个元组，关系 S 中有 3 个属性和 5 个元组，则  $R \times S$  的属性个数和元组个数分别是 B。
- A. 7 和 8 B. 7 和 15 C. 12 和 8 D. 12 和 15
14. 关系数据模型上的关系运算分为 B。
- A. 关系代数和集合运算 B. 关系代数和关系演算
- C. 关系演算和谓词演算 D. 关系代数和谓词演算
15. 在学生表（学号，姓名，性别）中，规定学号值域是 8 个数字组成的字符串，其规则属于 C。
- A. 实现完整性约束 B. 参照完整性约束

C. 用户自定义完整性约束      D. 关键字完整性约束

16. 能够把关系 R 和 S 进行自然连接时舍弃的元组放到结果关系中的操作是 D。

A. 左外连接      B. 右外连接      C. 外部并      D. 外连接

17. 五种基本关系代数运算是 A。

A.  $\cup$ ,  $-$ ,  $\times$ ,  $\pi$  和  $\sigma$       B.  $\cup$ ,  $-$ ,  $\bowtie$ ,  $\pi$  和  $\sigma$

C.  $\cup$ ,  $\cap$ ,  $\times$ ,  $\pi$  和  $\sigma$       D.  $\cup$ ,  $\cap$ ,  $\bowtie$ ,  $\pi$  和  $\sigma$

18. 设有关系模式 EMP (职工号, 姓名, 年龄, 技能), 假设职工号惟一, 每个职工有多项技能, 则 EMP 表的主键是 D。

A. 职工号      B. 姓名, 技能      C. 技能      D. 职工号, 技能

### 三、简答题

1. 名词解释:

属性, 域, 元组, 候选键, 主键, 外键。

2. 试阐述关系模型的 3 个组成部分。

3. 试阐述关系模型的完整性规则。在参照完整性中, 为什么外键属性的值也可以为空? 什么情况下不可以为空?

答: 外键不是候选键, 是可以为空的, 参照完整性只是要求当其有值时, 其值必须是相应主键的某个值。但当外键属性是主属性时, 即其是构成候选键的属性时, 根据实体完整性, 其值是不允许为空的。

4. 关系代数的基本运算有哪些? 如何用这些基本运算来表示其他运算?

答: 关系代数运算中的 5 种基本的操作包括: 并、差、笛卡尔积、投影和选择。

交:  $R \cap S = R - (R - S)$        $R \cap S = S - (S - R)$

连接:  $\pi_{K_1+K_2-K_3} \sigma_{R.K_3=S.K_3} (R \times S)$  关系 R 和 S 分别有  $k_1$  和  $k_2$  个属性, 有  $k_3$  个相同的属性

除:  $R \div S = \pi_X (R) - \pi_X ((\pi_X (R) \times S) - R)$

5. 试阐述关系的笛卡尔积、等值连接和自然连接的区别和联系。

6. 假设 R(a,b) 和 S(c,d), 试把如下元组演算表达式用关系代数表达式表示:

$$\{ t \mid R(t) \wedge (\exists u)(S(u) \wedge u[1] \neq t[2]) \}$$

答:  $\pi_{a,b} \sigma_{R.b=S.c} (R \times S)$

7. 关系 R 和 S 的半连接 (semijoin) 写作  $R \ltimes S$ , 它表示由 R 中的满足如下条件的元组 t 组成的集合: t 至少跟 S 中的一个元组在 R 和 S 的公共属性上相同。用三种不同的关系代数表达式给出  $R \ltimes S$  的等价表示。

答：关系 R 和 S 分别有  $k_1$  和  $k_2$  个属性，有  $k_3$  个相同的属性

$$\pi_{K_1} \sigma_{R.K_3=S.K_3} (R \times S)$$

$$\pi_{K_1} R \bowtie_{R.K_3=S.K_3} S$$

$$\pi_{K_1} R \bowtie S$$

8. 设有关系 R、S，如图 3-8。求  $\pi_c(S)$ 、 $\sigma_{B<'c'}(R)$ 、 $\sigma_{A=C}(R \times S)$ 、 $R \bowtie S$ 、 $R \bowtie_{R.B<S.B} S$

关系 R		关系 S	
A	B	B	C
a	b	b	c
c	b	e	a
d	e	b	d

图 3-8 关系 R 和 S 的模式及实例

答：注意选择和投影后剩余元组不能相同。

#### 四、查询实现题

1. 设有学生选课数据库，包括 S、C 和 SC 三个关系模式：

S (SNO, SN, SD, SB, SEX);

C (CNO, CN, PCNO, TN);

SC (SNO, CNO, GRADE);

学生表 S 包含学号 (SNO)、学生姓名 (SN)、所在系 (SD)、出生年月 (SB)、性别 (SEX) 等属性；

课程表 C 包含课程号 (CNO)、课程名 (CN)、先修课程号 (PCNO)、主讲教师姓名 (TN) 等属性；

选课表 SC 包含学号 (SNO)、课程号 (CNO)、成绩 (GRADE) 等属性。

试用关系代数、元组演算完成如下查询：

(1) 检索以“C2”课程作为先修课程的课程号；

$$\Pi \text{ CNO}(\sigma \text{ PCNO}='C2')(C)$$

$$\{t^{(1)} | (\exists v)(C(v) \wedge v[3] = 'C2' \wedge t[1] = v[1])\}$$

(2) 检索选修课程名为“Math”的学生学号与姓名；

$$\Pi \text{ SNO, SN}(\sigma \text{ CN}='Math')(S \bowtie SC \bowtie C)$$

$$\{t^{(2)} | (\exists u)(\exists v)(\exists w)(S(u) \wedge SC(v) \wedge C(w) \wedge u[1] = v[1] \wedge v[2] = w[1] \wedge w[2] = 'Math' \wedge t[1] = u[1] \wedge t[2] = u[2])\}$$



(3) 检索选修课程号为“C2”或“C4”课程的学生学号与成绩;

$$\Pi_{SNO, GRADE} (\sigma_{CNO='C2' \vee CNO='C4'}(SC))$$

$$\{t^{(2)} | (\exists v)(SC(v) \wedge (v[2]='C2' \vee v[2]='C4') \wedge t[1]=v[1] \wedge t[2]=v[3])\}$$

(4) 检索至少选修了课程号为“C2”和“C4”课程的学生姓名;

$$\Pi_{SN} ((\Pi_{SNO} (\sigma_{CNO='C2'}(SC)) \cap \Pi_{SNO} (\sigma_{CNO='C4'}(SC))) \bowtie S)$$

$$\{t^{(1)} | (\exists u)(\exists v)(\exists w)(SC(u) \wedge SC(v) \wedge S(w) \wedge u[1]=w[1] \wedge v[1]=w[1] \wedge u[2]='C2' \wedge v[2]='C4' \wedge t[1]=w[2])\}$$

(5) 检索选修课程包括“李斯”老师所授课程之一的学生学号;

$$\Pi_{SNO} (\Pi_{CNO} (\sigma_{TNO='李斯'}(C)) \bowtie SC)$$

$$\{t^{(1)} | (\exists u)(\exists v) (SC(u) \wedge C(v) \wedge u[2]=v[1] \wedge v[4]='李斯' \wedge t[1]=u[1])\}$$

(6) 检索全部学生都选修的课程的课程号和课程名;

$$\Pi_{CNO, CN} ((\Pi_{SNO, CNO}(SC) \div \Pi_{SNO}(S)) \bowtie C)$$

$$\{t^{(2)} | (\forall u)(\exists v)(\exists w)(S(u) \wedge C(v) \wedge SC(w) \wedge u[1]=w[1] \wedge v[1]=w[2] \wedge t[1]=v[1] \wedge t[2]=v[2])\}$$

(7) 检索计算机系学生都学习过的课程的课程号;

$$\Pi_{SNO, CNO} (\sigma_{SD='计算机'}(S) \bowtie SC) \div \Pi_{SNO} (\sigma_{SD='计算机'}(S))$$

$$\Pi_{SNO, CNO}(SC) \div \Pi_{SNO} (\sigma_{SD='计算机'}(S))$$

$$\{t^{(1)} | (\exists u)(SC(u) \wedge (\forall v)(S(v) \wedge (v[3]='计算机系' \rightarrow (\exists w)(SC(w) \wedge w[1]=v[1] \wedge w[2]=u[2]))) \wedge t[1]=u[2])\}$$

$$\{t^{(1)} | (\exists u)(C(u) \wedge (\forall v)(S(v) \wedge (v[3]='计算机系' \rightarrow (\exists w)(SC(w) \wedge w[1]=v[1] \wedge w[2]=u[2]))) \wedge t[1]=u[2])\}$$

(8) 检索“张山”同学不学课程的课程号。

$$\Pi_{CNO}(C) - \Pi_{CNO} (\Pi_{SNO} (\sigma_{SN='张山'}(S)) \bowtie SC)$$

$$\{t^{(1)} | (\exists u)(C(u) \wedge (\forall v)(SC(v) \wedge (u[1]=v[2] \rightarrow (\exists w)(S(w) \wedge w[1]=v[1] \wedge w[2] \neq '张山')))) \wedge t[1]=u[2])\}$$

$$\{t^{(1)} | (\exists u)(C(u) \wedge (\forall v)(SC(v) \wedge (u[1]=v[2] \rightarrow \neg (\exists w)(S(w) \wedge w[1]=v[1] \wedge w[2]='张山')))) \wedge t[1]=u[2])\}$$

2. 设有 SPJ 数据库, 包括 S、P、J 和 SPJ 四个关系模式, 数据库模式及实例值如图 3-9 所示。其中

供应商表 S 由供应商代码 (SNO)、供应商姓名 (SNAME)、供应商状态 (STATUS)、

供应商所在城市 (CITY) 组成;

零部件表 P 由零部件代码(PNO)、零部件名(PNAME)、颜色(COLOR)、重量(WEIGHT)组成;

工程项目表 J 由工程项目代码(JNO)、工程项目名(JNAME)、工程项目所在城市(CITY)组成;

供应情况表 SPJ 由供应商代码 (SNO)、零部件代码 (PNO)、工程项目代码 (JNO)、供应数量 (QTY) 组成, 表示某供应商供应某种零部件给某项工程的数量为 QTY。

试用关系代数、元组演算完成如下查询:

(1) 检索上海厂商供应的所有零部件的号码;

$$\Pi PNO(\sigma \text{ city} = '上海' (S) \bowtie SPJ)$$
$$\{t^{(1)} \mid (\exists u) (\exists v) (S(u) \wedge SPJ(v) \wedge u[4] = '上海' \wedge u[1] = v[1] \wedge t[1] = v[2])\}$$

(2) 检索使用上海厂商供应的零部件的工程名称;

$$\Pi JNAME(\sigma \text{ city} = '上海' (S) \bowtie SPJ \bowtie J)$$
$$\{t(1) \mid (\exists u) (\exists v) (\exists w) (S(u) \wedge SPJ(v) \wedge J(w) \wedge u[4] = '上海' \wedge u[1] = v[1] \wedge w[1] = v[3] \wedge t[1] = w[2])\}$$

(3) 检索供应工程 J1 零部件 P1 的供应商的号码;

$$\Pi sno(\sigma Jno = 'J1' \wedge Pno = 'P1' (SPJ))$$
$$\{t(1) \mid (\exists u) (SPJ(u) \wedge u[3] = 'J1' \wedge u[2] = 'P1' \wedge t[1] = u[1])\}$$

(4) 检索供应工程 J1 零部件为红色的供应商的号码;

$$\Pi sno(\sigma Jno = 'J1' \wedge Color = '红色' (SPJ \bowtie P))$$
$$\Pi sno(\sigma Color = '红色' (P) \bowtie (\sigma Jno = 'J1' (SPJ)))$$
$$\{t(1) \mid (\exists u) (\exists v) (SPJ(u) \wedge P(v) \wedge u[3] = 'J1' \wedge u[2] = v[1] \wedge v[3] = '红色' \wedge t[1] = u[1])\}$$

(5) 检索工程项目 J2 使用的各种零部件的名称及其数量;

$$\Pi PNAME, QTY(\sigma JNO = 'J2' (SPJ) \bowtie P)$$
$$\{t(2) \mid (\exists u) (\exists v) (SPJ(u) \wedge P(v) \wedge u[3] = 'J2' \wedge u[2] = v[1] \wedge t[1] = v[2] \wedge t[2] = u[4])\}$$

(6) 检索没有使用天津厂商供应的零部件的工程号码;

$$\Pi Jno(J) - \Pi Jno(\Pi Sno(\sigma CITY = '天津' (S)) \bowtie SPJ)$$
$$\{t(1) \mid (\exists u) (J(u) \wedge (\forall v) (SPJ(v) \wedge (u[1] = v[3] \rightarrow (\exists w) (S(w) \wedge w[1] = v[1] \wedge w[4] \neq '天津')))) \wedge t[1] = u[1])\}$$

$\{t(1) \mid (\exists u) (J(u) \wedge (\forall v) (SPJ(v) \wedge (u[1]=v[3] \rightarrow \neg (\exists w) (S(w) \wedge w[1]=v[1] \wedge W[4]='天津')))) \wedge t[1]=u[1])\}$

(7) 检索没有使用天津供应商供应的红色零部件的工程号;

$\Pi Jno(J) - \Pi Jno(\sigma_{CITY='天津' \wedge Color='红色'}(S \bowtie SPJ \bowtie P))$

$\Pi Jno(J) - \Pi Jno(\sigma_{CITY='天津'}(S) \bowtie SPJ \bowtie Color='红色'(P))$

$\{t(1) \mid (\exists u) (J(u) \wedge (\forall v) (SPJ(v) \wedge (u[1]=v[3] \rightarrow \neg ((\exists w) (S(w) \wedge w[1]=v[1] \wedge W[4]='天津') \wedge (\exists x) (P(x) \wedge x[1]=v[2] \wedge x[3]='红色')))) \wedge t[1]=u[1])\}$

例：对于 J 表中的每一个 JNO（比如 J1）去判断 SPJ 表中所有 JNO 为 J1 的元组，是否不存在其 SNO 对应 CITY 是“天津”，而且 PNO 对应颜色是红色，由于存在(S1,P1,J1,200)，所以 J1 不满足条件）。同理只有 J2,J5 满足条件。

$\{t(1) \mid (\exists u) (J(u) \wedge (\forall v) (SPJ(v) \wedge (u[1]=v[3] \rightarrow ((\exists w) (S(w) \wedge w[1]=v[1] \wedge W[4] \neq '天津') \vee (\exists x) (P(x) \wedge x[1]=v[2] \wedge x[3] \neq '红色')))) \wedge t[1]=u[1])\}$

例：对于 J 表中的每一个 JNO（比如 J2）去判断 SPJ 表中所有 JNO 为 J2 的元组，是否存在 SNO 对应 CITY 不是“天津”，或 PNO 对应颜色不是“红色”。虽然存在(S1,P2,J2,100) 天津供应商，(S5,P6,J2,200) 红色零件，但 J2 满足条件。而对于 J1，存在两个条件均不满足的元组，所以 J1 不满足条件。

$\{t(1) \mid (\exists x) (SPJ(x) \wedge \neg ((\exists u) (\exists v) (\exists w) ((S(u) \wedge SPJ(v) \wedge P(w) \wedge u[4]='天津' \wedge P[3]='红色' \wedge u[1]=v[1] \wedge v[2]=W[1] \wedge x[3]=v(3)))) \wedge t[1]=x[3])\}$

对于没有使用任何零件的工程，不能在结果中得到，错误。

$\{t(1) \mid (\exists u) (J(u) \wedge (\forall v) (SPJ(v) \wedge (u[1]=v[3] \rightarrow (\exists w) (\exists x) (S(w) \wedge P(x) \wedge w[1]=v[1] \wedge W[4] \neq '天津' \wedge x[1]=v(2) \wedge x[3] \neq '红色')))) \wedge t[1]=u[1])\}$  （有问题）

比如：由于存在 (S1,P2,J2,100)，所以 J2 不满足条件，错误。

(8) 检索至少使用了供应商 S1 所供应的全部零部件的工程号 Jno。

$\Pi Pno,Jno(SPJ) \div \Pi Pno(\sigma_{Sno='S1'}(SPJ))$

注：检索出的工程使用的零件包括所有供应商所供应的、具有与 S1 所供应零件相同的零件。

$\Pi Sno,Pno,Jno(SPJ) \div \Pi Sno,Pno(\sigma_{Sno='S1'}(SPJ))$

注：检索出的工程使用的零件包括由 S1 供应给其他工程的所有零件。

$\{t(1)|(\exists u)(SPJ(u) \wedge (\forall v)(SPJ(v) \wedge (v[1]=S1' \rightarrow (\exists w)(SPJ(w) \wedge w[2]=v[2] \wedge w[3]=u[3]))) \wedge t[1]=u[3])\}$

$\{t(1)|(\exists u) (SPJ(u) \wedge (\forall v)(SPJ(v) \wedge (v[1]=S1' \rightarrow (\exists w)(SPJ(w) \wedge w[1]=v[1] \wedge w[2]=v[2] \wedge w[3]=u[3]))) \wedge t[1]=u[3])\}$

或

$\{t(1)|(\exists u)(J(u) \wedge (\forall v)(SPJ(v) \wedge (v[1]=S1' \rightarrow (\exists w)(SPJ(w) \wedge w[2]=v[2] \wedge w[3]=u[1]))) \wedge t[1]=u[1])\}$

$\{t(1)|(\exists u) (J(u) \wedge (\forall v)(SPJ(v) \wedge (v[1]=S1' \rightarrow (\exists w)(SPJ(w) \wedge w[1]=v[1] \wedge w[2]=v[2] \wedge w[3]=u[1]))) \wedge t[1]=u[1])\}$

注：解答的结果中，如果 S1 没有对各工程提供任何零件，结果就是所有的工程号。

## 习 题

### 一、填空题

1. 在 SQL 语言中, 删除表 S 的语句是 **DROP TABLE S**。
2. SQL 语言使用 **CREATE VIEW** 语句建立视图。
3. 在 SQL 中, 用 **MAX()** 函数求最大值; 可以利用 **COUNT()** 函数计算记录的个数; 用 **AVG()** 函数求平均值。
4. 在 SQL 语言的关系定义语句中, 外键子句实现 **参照** 完整性约束; 主键实现 **实体** 完整性约束。
5. SQL 语言支持关系数据库三级模式结构, 其中外模式对应于 **视图**, 模式对应于 **基本表**, 内模式对应于存储文件。

### 二、选择题

1. 如果要修改表的结构, 应该使用的 SQL 语句是 **D**。  
A. UPDATE TABLE                      B. MODIFY TABLE  
C. CHANGE TABLE                      D. ALTER TABLE
2. 在数据库系统的三级模式结构中, 定义视图的组织方式属于 **B**。  
A. 概念模式              B. 外模式              C. 逻辑模式              D. 内模式
3. 在数据库系统中, 视图可以提供数据的 **C**。  
A. 完整性              B. 并发性              C. 安全性              D. 可恢复性
4. 可以在 SQL 查询时去掉重复数据的是 **D**。  
A. ORDER BY              B. DESC              C. GROUP BY              D. DISTINCT
5. 下列 SQL 语句中, 用来创建关系表的是 **B**。  
A. ALTER              B. CREATE              C. UPDATE              D. INSERT
6. 数据库管理系统能实现创建表, 删除表等操作的数据库语言称为 **A**。  
A. 数据定义语言 (DDL)              B. 数据管理语言  
C. 数据操纵语言 (DML)              D. 数据控制语言
8. 在 SQL 中, 可以用谓词 EXISTS 来测试一个集合中是否 **D**。  
A. 有相同元组存在              B. 为空集合  
C. 有相同分量存在              D. 为非空集合
9. 下列哪项不是视图机制的优点 **D**。  
A. 数据安全性              B. 逻辑独立性              C. 操作简便性              D. 数据完整性
10. SQL 的 SELECT 语句中, “HAVING 条件表达式”用来筛选满足条件的 **D**。  
A. 列              B. 行              C. 关系              D. 分组
12. 在 SQL 语言中, 属于 DML 的操作命令是 **C**。

A. CREATE            B. GRANT            C. UPDATE            D. DROP

13. SQL 语言的一次查询的结果是一个 **D**。

A. 数据项            B. 记录            C. 元组            D. 表

14. 如果想找出在关系 R 的 A 属性上不为空的那些元组,则选择子句应该为 **C**。

A. WHERE A!=NULL            B. WHERE A <> NULL  
C. WHERE A IS NOT NULL            D. WHERE A NOT IS NULL

15. 使用 SQL 语句在学生成绩表(学号, 课程号, 成绩)中, 查询平均成绩大于 60 分的学生时, 不必使用的子句是 **C**。

A. SELECT            B. GROUP BY            C. WHERE            D. HAVING

16. 下列哪组 SQL 命令全部属于数据定义语句的命令? **A**

A. CREATE, DROP, ALTER            B. CREATE, DROP, SELECT  
C. CREATE, DROP, GRANT            D. CREATE, DROP, UPDATE

18. 已知有关系模式 R (SNO, sname, age), 其中 SNO 表示学生的学号, 类型为 Char (8), 前 4 位表示入学年份。查询所有 2003 年入学的学生姓名 (sname), SQL 语句是 **B**。

A. SELECT sname FROM R WHERE SNO ='2003%'  
B. SELECT sname FROM R WHERE SNO LIKE '2003%'  
C. SELECT sname FROM R WHERE SNO ='2003\_'  
D. SELECT sname FROM R WHERE SNO LIKE '2003\_'

### 三、简答题

1. 什么是连接查询? 什么是嵌套查询? 什么是相关子查询?

答: 多个表之间存在某种联系, 如果一个查询要在两个以上的表中进行, 我们称这种查询为连接查询。

在子查询语句中再嵌入其他子查询, 这就是嵌套查询。

子查询结果依赖父查询的当前元组的属性值, 这种查询就是相关子查询。

2. 试阐述 SQL 的特点。

答: 1) .支持关系数据库系统的三级模式结构。

2) .语言功能强大。

3) .用户性能好。

4) .提供两种用户使用方式。

5) .高度非过程化。

3. 试介绍一个你所熟悉的 DBMS 产品的完整性功能。

4. 什么是基本表? 什么是视图? 两者的区别和联系是什么?

答: 视图是从一个或几个基本表导出的虚表, 视图为用户提供数据逻辑独立性等作用。

对视图的操作由系统自动转换成等价的对基本表的操作。

5. 在 SQL 中有哪些方法可以作表间连接, 请说明之, 并各举一例。

答: 多表连接、嵌套连接、集合操作等。

6. 视图有哪些优点?

答: 1) 视图提供了一个简化用户操作的快捷方式。

2) 视图支持多用户同时以不同的方式对相同的数据进行查询。

3) 视图对于隐藏的数据自动提供安全保护。

4) 视图可以为用户和应用程序提供逻辑上的数据独立性。

7. 所有的视图是否都可以更新? 为什么?

答: 不是所有的视图都是可更新的, 有些对视图的更新操作不能唯一地有意义地转换成对相应基本表的更新, 就不能进行更新。

8. 试解释触发器和完整性约束之间的关系。

答: 触发器能实现完整性约束, 同时能够进行比完整性约束更为复杂的检查和操作。

9. 已知 R 和 S 两个关系如图所示:

R			S		
A	B	C	C	D	E
a1	b1	c1	c1	d1	e1
a2	b2	c2	c2	d2	e2
a3	b3	c3	c3	d3	e2

执行如下 SQL 语句, 给出执行结果。

```
(1) CREATE VIEW H (A, B, C, D, E)
      AS SELECT A, B, R.C, D, E
      FROM R, S
      WHERE R.C=S.C;
```

```
(2) SELECT B, D, E
      FROM H
      WHERE C='c2';
```

(1)

A	B	C	D	E
a1	b1	c1	d1	e1
a2	b2	c2	d2	e2
a3	b3	c3	d3	e3

(2)

B	D	E
b2	d2	e2

10. 已知某公司数据库包含如下四个基本表，

Department (Dept\_No, Dept\_Name, Location)

Employee (Emp\_No, Emp\_Name, Dept\_no)

ProJect (Pro\_No, Pro\_Name, Budget)

Works (Emp\_No, Pro\_No, Job)

(1) 使用DDL语句定义上述四个表，并说明主键和外键。

(2) 将基本表Works中Job = ‘经理’的员工号及参加的项目号，定义为一个视图V\_MANAGER( Emp\_No, Pro\_No)。

(3) 将“P2”项目的所有员工号、员工名和所在部门名称，定义为一个视图V\_P2(Emp\_No, Emp\_Name, Dept\_Name, )。

解答：

(1) CREAT TABLE Deparment

(Dep\_NO CHAR (10) ,

Dep\_NAME CHAR (20) ,

Location CHAR (30) ,

PRIMARY KEY(Dep\_NO));

CREAT TABLE Employee

(Emp\_NO CHAR(10) PRIMARY KEY,

Emp\_NAME CHAR(8),

Dep\_NO CHAR(10),

FOREIGN KEY (Dep\_NO) REFERENCES Department(Dept\_NO));

CREATE TABLE ProJect

(Pro\_NO CHAR(10) PRIMARY KEY,

Pro\_NAME CHAR(20),

Budget INT)

CREATE TABLE Works

(Emp\_NO CHAR(10),

Pro\_NO CHAR(10),

Job CHAR(20),



```
PRIMARY KEY(Emp_NO,Pro_NO),
FOREIGN KEY(Emp.NO) REFERENCES Employee(Emp_NO),
FOREIGN KEY(Pro_NO) REFERENCES ProJect(Pro_NO));
```

```
(2)CREAT VIEW V_MANAGER (Emp_NO,Pro-NO)
AS SELECT EMP_NO,Pro-NO
FROM(WORKS)
WHERE JoB='经理';
```

```
(3)CREATE VIEW V_P2(Emp_NO,Emp_NAME,Dept_NAME)
AS SELECT Employee.Emp_NO,
Employee.Emp_NAME,
Department.Dept_NAME
FROM Department,Employee, works
WHERE WORKS..Pro_no ='P2' AND
Employee.Emp_NO= works.Emp_NO AND
Department. Dep_NO= Employee. Dep_NO;
```

#### 四、查询实现题

1. 针对第3章课后习题查询实现部分的第2小题所给出的SPJ数据库，试用SQL语句完成如下操作：

- (1) 求供应工程 J1 零部件的供应商号码 SNO；
- (2) 求供应工程 J1 零部件 P1 的供应商名称；
- (3) 求供应工程 J1 红色零部件的供应商号码 SNO；
- (4) 求没有使用天津供应商供应的红色零部件的工程号 JNO；
- (5) 求至少用了供应商 S1 所供应的全部零部件的工程号 JNO。
- (6) 找出使用供应商 S1 所供应零部件的工程号码；
- (7) 找到工程项目 J2 使用的各种零部件的名称及其数量；
- (8) 找出上海厂商供应的所有零部件的号码；
- (9) 找出使用上海厂商供应的零部件的工程名称；
- (10) 求供应商 S2 供应给各个工程的零部件总数量，结果中要有所对应的工程名称。
- (11) 把全部红色零部件的颜色改成蓝色。
- (12) 为工程项目名称为“一汽”的工程建立一个供应情况的视图，包括供应商代码 SNO、

零部件代码 Pno、供应数量 Qty 。

解答：

(1) 求供应工程 J1 零部件的供应商号码 SNO;

Select SNO

From SPJ

Where JNO='J1';

(2) 求供应工程 J1 零部件 P1 的供应商名称;

Select SNAME

From SPJ, S

Where SPJ.JNO='J1' AND SPJ.Pno='P1' AND S.SNO= SPJ.Sno;

(3) 求供应工程 J1 红色零部件的供应商号码 SNO;

Select SNO

Form SPJ, P

Where SPJ.Pno=P.Pno and

JNO='J1' and

Color='红'

(4) 求没有使用天津供应商供应的红色零部件的工程号 JNO

Select JNO

Form J

Where JNO not IN

(select JNO

From S,SPJ,P

Where SPJ.Pno=P.Pno and

SPJ.SNO=S.SNO and

City='天津' and

Color='红');

(5) 求至少用了供应商 S1 所供应的全部零部件的工程号 JNO。

Select JNO

From J

Where not exists

( select \*

Form SPJ as SPJ1

Where SNO='s1' and not exists

( select \*

From SPJ as SPJ2

Where SPJ2.SNO='s1' and

SPJ2.Pno=SPJ1.Pno and

SPJ2.JNO=J.JNO) );

注：条件 SPJ2.SNO='s1'也可不要。

(6) 找出使用供应商 S1 所供应零部件的工程号码;

```
Select JNO
From SPJ
Where SNO='S1';
```

(7) 找到工程项目 J2 使用的各种零部件的名称及其数量;

```
Select Pname,Qty
From SPJ,P
Where SPJ.Pno=P.Pno and JNO='J2';
```

(8) 找出上海厂商供应的所有零部件的号码;

```
Select Pno
Form SPJ,S
Where SPJ.SNO=S.SNO and S.city='上海';
```

(9) 找出使用上海厂商供应的零部件的工程名称;

```
Select Jname
From SPJ,J,S
Where SPJ.SNO=s.SNO and
SPJ.JNO=J.JNO and
S.city='上海';
```

(10) 求供应商 S2 供应给各个工程的零部件总数量, 结果中要有所对应的工程名称。

```
Select Jname, Sum(Qty)
From SPJ,J
Where SNO='S2' and SPJ.JNO=J.JNO
Group by J. JNO, Jname;
```

(11) 把全部红色零部件的颜色改成蓝色。

```
Update P
Set color='蓝'
Where color='红';
```

(12) 为工程项目名称为“一汽”的工程建立一个供应情况的视图, 包括供应商代码 SNO、零部件代码 Pno、供应数量 Qty。

```
Create view 供应情况(SNO,Pno,Qty)
AS select SNO, Pno, Qty
From J,SPJ
Where J.JNO=SPJ.JNO AND Jname='一汽'
```

2. 设有学生选课数据库，包括 S、C 和 SC 三个关系模式：

S (SNO, SN, SD, SB, SEX)

C (CNO, CN, PC, TN)

SC (SNO, CNO, GRADE)

试用 SQL 语句实现下列操作：

- (1) 用连接，嵌套和 EXISTS 查询来查询学习课程号为“C01”的学生学号和姓名；
- (2) 查询至少选修课程号为“C03”和“C04”的学生姓名；
- (3) 查询不学“C02”课程的学生姓名和性别；
- (4) 查询学习全部课程的学生姓名；
- (5) 查询所学课程包含学生“李珊”所学课程的学生的学号。
- (6) 在关系 C 中插入一个元组 (‘C08’, ‘VB’, ‘C02’, ‘黄萍’);
- (7) 查询平均成绩大于 80 分的课程任课老师的姓名，并将查询到的值送入另一个已经存在的表 LEVEL\_80 (TNAME) 中；
- (8) 删除 SC 中还没有成绩的元组；
- (9) 删除选修“李”老师课程的所有女生选课元组；
- (10) 把数学成绩小于 60，但大于 55 分的同学的成绩全改为 60 分；
- (11) 在表 SC 中，当某个成绩低于某一门课程的平均成绩时，提高 5%。
- (12) 建立视图 V\_SSC(SNO, SN, CNO, CN, G)，并按 CNO 升序排列；
- (13) 从视图 V\_SSC 上查询平均成绩在 90 分以上的学生的 SN, CN, G。
- (14) 授权用户张山对表 S, C 的查询权限；
- (15) 授权用户李斯对表 SC 的插入和删除权限；
- (16) 授权用户王武对表 SC 具有查询权限，对成绩属性具有更新权限；然后再撤销其对成绩属性所具有的更新权限；
- (17) 授权用户赵亮对所有表具有所有权限，并具有给其他用户授权的权限；
- (18) 创建一个视图，包含每门课程的课程号，课程的最高成绩、最低成绩、平均成绩；再创建一个具有查询该视图权限的角色，并将角色权限授权给用户周力。

解答：

- (1) 用连接，嵌套和 EXISTS 查询来查询学习课程号为“C01”的学生学号和姓名；

连接：

```
SELECT  S.SNO,S.SN
      FROM  S,SC
      WHERE  S.SNO=SC.SNO AND SC.CNO='C01';
```

嵌套：

```
SELECT SNO, SN
```

```

FROM S
WHERE SNO IN (SELECT SNO FROM SC
WHERE CNO='C01');

```

EXISTS 查询:

```

SELECT SNO,SN
FROM S
WHERE EXISTS (SELECT * FROM SC
WHERE S.SNO=SC.SNO AND SC.CNO='C01');

```

(2) 查询至少选修课程号为“C03”和“C04”的学生姓名;

```

SELECT SN
FROM S
WHERE SNO IN
(SELECT SC1.SNO
FROM SC SC1,SC SC2
WHERE SC1.CNO='C03' AND
SC2.CNO='C04'AND
SC1.SNO=SC2.SNO);

```

(3) 查询不学“C02”课程的学生姓名和性别;

```

SELECT SN,SEX
WHERE SNO NOT IN
(SELECT SNO
FROM SC
WHERE CNO='C02');

```

(4) 查询学习全部课程的学生姓名;

方法一:

```

SELECT SN
FROM S
WHERE not EXISTS
(SELECT *
FROM C
WHERE CNO not IN
(SELECT CNO
FROM SC
WHERE SC.SNO=S.SNO AND SC.CNO=C.CNO));

```

方法二:

```

SELECT SN
FROM S
WHERE not EXISTS

```

```

(SELECT *
FROM C
WHERE not EXISTS
  (SELECT *
   FROM SC
   WHERE SC.SNO=S.SNO AND SC.CNO=C.CNO)) ;

```

方法三:

```

SELECT SN
FROM S
WHERE SNO IN
  (SELECT SNO
   FROM SC
   GROUP BY SNO
   HAVING COUNT(*)=(SELECT COUNT(*) FROM C));

```

(5) 查询所学课程包含学生“李珊”所学课程的学生的学号。

方法一:

```

SELECT DISTINCT SNO
FROM SC X
WHERE NOT EXISTS
  (SELECT *
   FROM SC Y, S
   WHERE S.SN='李珊' AND
         S.SNO=Y.SNO AND
         Y.CNO NOT IN
           (SELECT CNO
            FROM SC Z
            WHERE X.SNO=Z.SNO));

```

方法一:

```

SELECT DISTINCT SNO
FROM SC X
WHERE NOT EXISTS
  (SELECT *
   FROM SC Y, S
   WHERE Y.SNO=S.SNO AND
         S.SN='李珊' AND NOT EXISTS
           (SELECT *
            FROM SC Z
            WHERE Z.SNO=X.SNO AND Z.CNO=Y.CNO));

```

(6) 在关系 C 中插入一个元组 ('C08', 'VB', 'C02', '黄萍');

```
INSERT
  INTO C
  VALUES ('C08', 'VB', 'C02', '黄萍');
```

(7) 查询平均成绩大于 80 分的课程任课老师的姓名, 并将查询到的值送入另一个已经存在的表 LEVEL\_80 (TNAME) 中;

```
INSERT
  INTO LEVEL_80(TNAME)
  SELECT TN
  FROM C
  WHERE CNO IN
    (SELECT CNO
     FROM SC
     GROUP BY CNO
     HAVING AVG(GRADE)>80);
```

或

```
INSERT
  INTO LEVEL_80(TNAME)
  SELECT TN
  FROM SC,C
  WHERE C.CNO=SC.CNO
  GROUP BY TN, SC.CNO
  HAVING AVG(GRADE)>80;
```

(8) 删除 SC 中还没有成绩的元组;

```
DELETE
  FROM SC
  WHERE GRADE IS NULL;
```

(9) 删除选修“李”老师课程的所有女生选课元组;

```
DELECT
  FROM SC
  WHERE SNO IN
    (SELECT SNO
     FROM S
     WHERE SEX='女')
  AND CNO IN
    (SELECT CNO
```

FROM C

WHERE TN like '李%');

(10) 把数学成绩小于 60, 但大于 55 分的同学的成绩全改为 60 分;

UPDATE SC

SET GRADE=60

WHERE SNO IN

(SELECT SNO

FROM SC.C

WHERE SC.CNO=C.CNO AND CN='数学' AND GRADE<60 AND GRADE>55);

(11) 在表 SC 中, 当某个成绩低于某一门课程的平均成绩时, 提高 5%。

UPDATE SC

SET GRADE=GRADE\*(1+5%)

WHERE GRADE< ANY (SELECT AVG(GRADE)

FROM SC

GROUP BY CNO);

(12) 建立视图 V\_SSC(SNO, SN, CNO, CN, G), 并按 CNO 升序排列;

**CREAT VIEW V\_SSC(SNO, SN, CNO, CN, G)**

**AS SELECT TOP 100 PERCENT S.SNO, S.SN, C.CNO, C.CN, SC.GRADE**

**FROM S,C,SC**

**WHERE S.SNO=SC.SNO AND**

**SC.CNO=C.CNO**

**ORDOR BY CNO ASC;**

注: 在 SQL SERVER 中要加上 TOP 100 PERCENT 才能使用 ORDOR BY。

(13) 从视图 V\_SSC 上查询平均成绩在 90 分以上的学生的 SN, CN, G。

**SELECT SN,CN,G**

**FROM V\_SSC**

**WHERE SNO IN (SELECT SNO**

**FROM V\_SSC**

**GROUP BY SNO**

**HAVING AVG(G)> 90);**

(14) 授权用户张山对表 S, C 的查询权限;

GRANT SELECT

ON S,C

TO 张山;

(15) 授权用户李斯对表 SC 的插入和删除权限;

GRANT INSERT ,DELECT

ON SC



TO 李斯;

(16) 授权用户王武对表 SC 具有查询权限，对成绩属性具有更新权限；然后再撤销其对成绩属性所具有的更新权限；

```
GRANT SELECT,UPDATE(GRADE)
ON SC
TO 王五
REVOKE UPDATE(GRADE)
ON SC
FROM 王五
```

(17) 授权用户赵亮对所有表具有所有权限，并具有给其他用户授权的权限；

```
GRANT ALL PRIVILEGES ON ALL TABLES
TO 赵亮
WITH GRANT OPTION;
```

(18) 创建一个视图，包含每门课程的课程号，课程的最高成绩、最低成绩、平均成绩；再创建一个具有查询该视图权限的角色，并将角色权限授权给用户周力。

```
CREATE VIEW COURSE(CNO,MAX,MIN,AVG)
AS SELECT CNO, MAX(GRADE), MIN(GRADE), AVG(GRADE)
FROM SC
GROUP BY CNO;
CREATE ROLE R1;
GRANT SELECT ON COURSE TO R1;
sp_addrolemember R1,周力;
```

3. 设有车辆管理数据库的数据模式如下：

车辆（车号，车牌名，车颜色，生产厂名）

工厂（厂名，厂长姓名，所在城市名）

城市（城市名，人口，市长姓名）

请用 SQL 语句实现如下查询：

- (1) 查询所有车牌名为“红旗”的轿车的车号。
- (2) 查询“红旗”牌轿车的生产厂家及厂长姓名。
- (3) 查询“跃进”牌轿车的生产厂家及所在城市的市长姓名。
- (4) 查询第一汽车制造厂所生产车辆的颜色。
- (5) 查询武汉生产哪些品牌的车。

解答：

- (1) 查询所有车牌名为“红旗”的轿车的车号。

```
SELECT 车号  
FROM 车辆  
WHERE 车牌名='红旗';
```

(2) 查询“红旗”牌轿车的生产厂家及厂长姓名。

```
SELECT 生产厂名, 厂长姓名  
FROM 车辆, 工厂  
WHERE 生产厂名=厂名 AND 车牌名='红旗';
```

(3) 查询“跃进”牌轿车的生产厂家及所在城市的市长姓名。

```
SELECT 生产厂名, 市长姓名  
FROM 车辆, 工厂, 城市  
WHERE 生产厂名=厂名 AND  
所在城市名=城市名 AND  
车牌名='跃进';
```

(4) 查询第一汽车制造厂所生产车辆的颜色。

```
SELECT DISTINCT 车颜色  
FROM 车辆  
WHERE 生产厂名='第一汽车制造厂';
```

(5) 查询武汉生产哪些品牌的车。

```
SELECT DISTINCT 车牌名  
FROM 车辆, 工厂  
WHERE 生产厂名=厂名 AND  
所在城市名='武汉';
```

## 习 题

### 一、填空题

1. 数据冗余不仅浪费存储空间,还可能导致操作异常和 数据不一致。操作异常包括 插入异常、修改异常 和 删除异常。
2. 满足 1NF、2NF 和 3NF 的模式集合之间,是一种 包含 关系。
3. 在函数依赖的范围内, BC NF 达到了最高的规范化程度。
4. 若关系 R 所有的属性都是不可再分的数据项,则称 R 属于第 二 范式。
5. 1NF 的关系消除非主属性对候选键的 部分 函数依赖后,可将范式等级提高到 2NF。
6. 关系模式由 2NF 转化为 3NF 是消除了非主属性对码的 传递函数依赖。
7. 关系模式由 3NF 转化为 BCNF 是消除了 主属性对于候选键的部分和传递函数依赖。
8. F 中的函数依赖所蕴含的函数依赖的集合称为 F 的 闭包, Armstrong 公理 为计算  $F^+$  提供了一个有效且完备的基础理论。
9. 当且仅当两个函数依赖集的 闭包 相等时,这两个函数依赖集等价。
10. 关系模式的规范化是通过 模式分解 来实现的。

### 二、选择题

1. 关系数据库规范化理论是为解决关系数据库中 A 问题而引入的。  
A. 插入、修改、删除异常和数据冗余    B. 提高查询速度  
C. 减少数据操作的复杂性    D. 保证数据的安全性和完整性
2. 设有关系模式 EMP (职工号, 姓名, 年龄, 技能)。假设职工号惟一, 每个职工有多项技能, 则 EMP 表的主键是 D。  
A. 职工号    B. 姓名, 技能    C. 技能    D. 职工号, 技能
3. 设关系模式 R (A, B) 上的函数依赖为  $A \rightarrow B$ , 则 R 最高属于 C—D。  
A. 2NF    B. 3NF    C. BCNF    D. 4NF
4. 3NF D 规范为 4NF。  
A. 消除非主属性对码的部分函数依赖  
B. 消除非主属性对码的传递函数依赖  
C. 消除主属性对码的部分和传递函数依赖  
D. 消除非平凡且非函数依赖的多值依赖
5. 关系模式 R 中若没有非主属性, 则 B。  
A. R 属于 2NF 但不一定属于 3NF    B. R 属于 3NF 但不一定属于 BCNF  
C. R 属于 BCNF 但不一定属于 4NF    D. R 属于 4NF

6. 设  $R(U)$  是属性集  $U$  上的关系模式。 $X$ 、 $Y$  是  $U$  的子集。若对于  $R(U)$  的任意一个可能的关系  $r$ ， $r$  中不可能存在两个元组在  $X$  上的属性值相等，而在  $Y$  上的属性值不等，则称 A。

- A.  $Y$  函数依赖于  $X$                       B.  $Y$  对  $X$  完全函数依赖  
C.  $X$  为  $U$  的候选码                      D.  $R$  属于 2NF

7. 具有多值依赖的关系模式仍存在如下问题 D。

- A. 插入异常                      B. 删除异常                      C. 数据冗余太大  
D. 插入异常、删除异常、数据冗余太大

8. 当  $B$  属性函数依赖于  $A$  属性时，属性  $A$  与  $B$  的联系是 B。

- A. 1 对多                      B. 多对 1                      C. 多对多                      D. 以上都不是

9. 关系模式分解的无损连接和保持函数依赖两个特性之间 D。

- A. 前者蕴含后者                      B. 后者蕴含前者  
C. 二者同时成立，或同时不成立                      D. 没有必然联系

10. 设关系模式  $R(A, B, C, D)$ ， $F$  是  $R$  上成立的 FD 集， $F = \{AB \rightarrow C, D \rightarrow B\}$ ，那么  $F$  在模式  $ACD$  上的投影  $\pi_{ACD}(F)$  为 C。

- A.  $\{AB \rightarrow C, D \rightarrow B\}$                       B.  $\{AC \rightarrow D\}$   
C.  $\{AD \rightarrow C\}$                       D.  $\Phi$  (即不存在非平凡的 FD)

### 三、简答题

1. 理解并给出下列术语的定义：

函数依赖、部分函数依赖、完全函数依赖、传递函数依赖、候选键、1NF、2NF、3NF、BCNF、多值依赖、4NF。

2. 建立一个关于系、学生、班级、学会等诸信息的关系数据库。

描述学生的属性有：学号、姓名、出生年月、系名、班号、宿舍区；

描述班级的属性有：班号、专业名、系名、人数、入校年份；

描述系的属性有：系名、系号、系办公室地点、人数；

描述学会的属性有：学会名、成立年份、地点、人数。

有关语义如下：一个系有若干专业，每个专业每年只招一个班，每班有若干学生。一个系的学生住在同一宿舍区。每个学生可参加若干学会，每个学会会有若干学生。学生参加某学会会有一个入会年份。

请给出关系模式，写出每个关系模式的最小函数依赖集，指出是否存在传递函数依赖，对于函数依赖左部是多属性的情况讨论函数依赖是完全函数依赖，还是部分函数依赖。指出各关系的候选键、外键，有没有全键存在？

解答：

按所提供的信息，可直接设计如下关系模式：

学生 (< 学号, 姓名, 出生年月, 系名, 班号, 宿舍区>, {学号→姓名, 学号→出生年月, 学号→班号, 班号→系名, 系名→宿舍区});

存在传递函数依赖：

学号→班号, 班号→系名, 系名→宿舍区

班级 (<班号, 专业名, 系名, 人数, 入校年份>, {班号→专业名, 专业名→系名, 班号→人数, 班号→入校年份, (专业名, 入校年份)→班号});

存在传递函数依赖：

班号→专业名, 专业名→系名;

系 (<系名, 系号, 系办公地点, 人数>, {系名→系号, 系号→系名, 系名→系办公地点, 系名→人数});

学会 (< 学会名, 成立年份, 地点, 人数>, {学会名→成立年份, 学会名→地点, 学会名→人数});

学生学会 (<学生, 学会, 入会时间>,{(学生, 学会)→入会时间}),

学生、学会是该模式的外码, (学生, 学会)→入会时间 是完全函数依赖。

标注红色的属性为主码，带下划线的为外码。

需要强调的是，由于学生和班级的关系模式中存在着传递函数依赖，所以这两个关系模式不满足 3NF，模式设计的不好，可采用分解的方法，将该两个模式规范化为 3NF，即可得到

学生 (< 学号, 姓名, 出生年月, 班号>, {学号→姓名, 学号→出生年月, 学号→班号 });

班级 (<班号, 专业名, 人数, 入校年份>, {班号→专业名, 班号→人数, 班号→入校年份, (专业名, 入校年份)→班号});

专业 (<专业名, 系名>,专业名→系名) ;

系 (<系名, 系号, 系办公地点, 人数, 宿舍区>, {系名→系号, 系号→

系名, 系名 $\rightarrow$ 系办公地点, 系名 $\rightarrow$ 人数, 系名 $\rightarrow$ 宿舍区});

通过该习题, 可以让学生体会规范化理论的指导意义。

3. 下面的结论哪些是正确的? 哪些是错误的? 对于错误的请给出一个反例说明。

(1) 任何一个二目关系是属于 3NF 的。

(2) 任何一个二目关系是属于 BCNF 的。

(3) 任何一个二目关系是属于 4NF 的。

(4) 当且仅当函数依赖  $A \rightarrow B$  在  $R$  上成立, 关系  $R(A,B,C)$  等于其投影  $R_1(A,B)$  和  $R_2(A,C)$  的连接。

(5) 若  $R.A \rightarrow R.B, R.B \rightarrow R.C$ , 则  $R.A \rightarrow R.C$ 。

(6) 若  $R.A \rightarrow R.B, R.A \rightarrow R.C$ , 则  $R.A \rightarrow R.(B,C)$ 。

(7) 若  $R.B \rightarrow R.A, R.C \rightarrow R.A$ , 则  $R.(B,C) \rightarrow R.A$ 。

(8) 若  $R.(B,C) \rightarrow R.A$ , 则  $R.B \rightarrow R.A, R.C \rightarrow R.A$ 。

解答:

(1) 正确, 只有两个属性, 无传递依赖。

(2) 正确, 只有两个属性, 根据定义, 不存在决定因素为非码的依赖。

(3) 正确, 只有两个属性, 所以无非平凡的多值依赖。

(4) **错误**。若函数依赖  $A \rightarrow B$  在  $R$  上成立, 因为  $U_1=AB, U_2=AC$ , 则  $U_1 \cap U_2=A, U_1-U_2=B, (U_1 \cap U_2) \rightarrow (U_1-U_2) \in F^+$ , 由教材定理 5.6,  $R_1(A,B)$  和  $R_2(A,C)$  是关系  $R(A,B,C)$  的无损分解, 因此关系  $R(A,B,C)$  等于其投影  $R_1(A,B)$  和  $R_2(A,C)$  的连接。

反之, 关系  $R(A,B,C)$  等于其投影  $R_1(A,B)$  和  $R_2(A,C)$  的连接, 即分解无损, 则  $(U_1 \cap U_2) \rightarrow (U_1-U_2) \in F^+$  或  $(U_1 \cap U_2) \rightarrow (U_2-U_1) \in F^+$ ,

即  $A \rightarrow B$  或  $A \rightarrow C$ 。

(5) 正确, 传递律

(6) 正确, 合并规则

(7) 正确, 增广律

(8) 错误, 例如:  $SC(SNO, CNO, GRADE), (SNO, CNO) \rightarrow GRADE$ , 但  $SNO \rightarrow GRADE, CNO \rightarrow GRADE$  不成立。

#### 四、计算题

1. 设关系模式  $R(U, F)$ , 其中,  $U = \{H, I, J, K, L, M\}$ ,  $F = \{H \rightarrow I, K \rightarrow H, LM \rightarrow K,$

设置了格式: 字体颜色: 红色

$I \rightarrow L, KH \rightarrow M$ 。求 R 的候选键。

解答：

J 为 N 类属性，必包含在任一候选码中。没有 L 类属性。根据候选码的定义：不存在其真子集决定全属性。可通过去掉 U 中的冗余属性，求得模式的候选码。（利用教材算法 5.2）

$$\text{Key} = \{H, I, J, K, L, M\}$$

$$\because \{ \text{Key} - H \}_F^+ = U \quad \therefore \text{Key} = \text{Key} - H = \text{JIKLM}$$

$$\because \{ \text{Key} - I \}_F^+ = U \quad \therefore \text{Key} = \text{Key} - I = \text{JKLM}$$

$$\because \{ \text{Key} - K \}_F^+ = U \quad \therefore \text{Key} = \text{Key} - K = \text{JLM}$$

$$\because \{ \text{Key} - L \}_F^+ = \{J, M\} \neq U \quad \therefore \text{该关键字中必定含有属性 } L$$

$$\because \{ \text{Key} - M \}_F^+ = \{J, L\} \neq U \quad \therefore \text{该关键字中必定含有属性 } M$$

最后得到该关系的一个关键字  $\{J, L, M\}$

同理，处理属性的顺序不同，可求得其他候选码 JK, JHM, JIM。

2. 设有关系模式 R (XYZ), F 是 R 上的 FD 集,  $F = \{XY \rightarrow Z, Z \rightarrow X\}$ 。R 被分解为  $\rho = \{R_1(XY), R_2(XZ)\}$ 。试判断该分解是否保持函数依赖？

解答：

(1) 首先根据分解的定义，确定分解后的函数依赖，然后再加以判断。利用 Armstrong 公理的传递律，可将  $\rho$  分解为

$$\rho = \{ R_1 < (XY) >, R_2 < (XZ) \}, \{ Z \rightarrow X \} >$$

注：  $R_1 < (XY) >$  上没有非平凡的函数依赖。

(2) 只需判断  $XY \rightarrow Z$  是否为 G 所覆盖即可

$$G = \{ Z \rightarrow X \}$$

求得  $(XY)_G^+ = \{XY\}$ , Z 不属于  $(XY)_G^+$ ,  $XY \rightarrow Z$  不为 G 所覆盖。

故  $\rho$  不保持函数依赖。

3. 关系模式 R(U, F), 其中  $U = \{C, T, H, I, S, G\}$ ,  $F = \{CS \rightarrow G, C \rightarrow T, TH \rightarrow I, HI \rightarrow C, HS \rightarrow I\}$ , 将其分解成满足 3NF 且具有无损连接性和保持函数依赖的关系模式。

此题与例题 5-28 重复了，且已满足 3NF，取消。

4. 有关系模式 R (U, F),  $U = \{A, B, C, D, E\}$ ,  $F = \{A \rightarrow C, C \rightarrow D, B \rightarrow C, DE \rightarrow C, CE \rightarrow A\}$ ,

(1) 给出 R 候选键；

(2) 判断  $\rho = \{AD, AB, BC, CDE, AE\}$  是否为无损连接分解；

(3) 将 R 分解为 BCNF，并使之具有无损连接性。

解答：

(1) 经判断，F 已为最小函数依赖集，BE 为 L 类属性，且  $BE^+_F = U$ ，所以 BE 为 R 的候选键。

(2) 首先构造一个二位表如下：

属性 模式	A	B	C	D	E
R1(AD)	A1	B12	B13	A4	B15
R2(AB)	A1	A2	B23	B24	B25
R3(BC)	B31	A2	A3	B34	B35
R4(CDE)	B41	B42	A3	A4	A5
R5(AE)	A1	B52	B53	B54	A5

对 F 进行扫描，对于  $A \rightarrow C$ ，表修改如下：

属性 模式	A	B	C	D	E
R1(AD)	A1	B12	B13	A4	B15
R2(AB)	A1	A2	B13	B24	B25
R3(BC)	B31	A2	A3	B34	B35
R4(CDE)	B41	B42	A3	A4	A5
R5(AE)	A1	B52	B13	B54	A5

对于  $C \rightarrow D$ ，表修改如下：

属性 模式	A	B	C	D	E
R1(AD)	A1	B12	B13	A4	B15
R2(AB)	A1	A2	B13	A4	B25
R3(BC)	B31	A2	A3	A4	B35
R4(CDE)	B41	B42	A3	A4	A5



R5(AE)	A1	B52	B13	A4	A5
--------	----	-----	-----	----	----

对于  $B \rightarrow C$ ，表修改如下：

属性 模式	A	B	C	D	E
R1(AD)	A1	B12	B13	A4	B15
R2(AB)	A1	A2	A3	A4	B25
R3(BC)	B31	A2	A3	A4	B35
R4(CDE)	B41	B42	A3	A4	A5
R5(AE)	A1	B52	B13	A4	A5

对于  $DE \rightarrow C$ ，表修改如下：

属性 模式	A	B	C	D	E
R1(AD)	A1	B12	A3	A4	B15
R2(AB)	A1	A2	A3	A4	B25
R3(BC)	B31	A2	A3	A4	B35
R4(CDE)	B41	B42	A3	A4	A5
R5(AE)	A1	B52	A3	A4	A5

对于  $CE \rightarrow A$ ，

属性 模式	A	B	C	D	E
R1(AD)	A1	B12	A3	A4	B15
R2(AB)	A1	A2	A3	A4	B25
R3(BC)	B31	A2	A3	A4	B35
R4(CDE)	A1	B42	A3	A4	A5
R5(AE)	A1	B52	A3	A4	A5

表有改动，循环扫描

对  $A \rightarrow C, C \rightarrow D, B \rightarrow C, DE \rightarrow C, CE \rightarrow A$  再次扫描, 表不改变, 算法结束。不存在一行全为 A, 所以分解是有损的。

(3) BE 为 R 的候选键。由此可判断 R 不属于 BCNF。

考虑  $A \rightarrow C$ , A 不是 R 的码, 将 R 分解为  $\rho = \{R1 \langle \{AC\}, \{A \rightarrow C\} \rangle, R2 \langle \{ABDE\}, \{A \rightarrow D, B \rightarrow D, \underline{BE \rightarrow A}\} \rangle\}$ ; ( $\because A \rightarrow C, B \rightarrow C, C \rightarrow D \therefore A \rightarrow D, B \rightarrow D, \underline{BE \rightarrow A}$ , 但依赖  $DE \rightarrow C, \underline{CE \rightarrow A}$  丢失)

R1 是 BCNF, R2 不是, 需进一步分解。

考虑  $A \rightarrow D$ , A 不是 R2 的码, 将 R2 分解为  $\rho = \{R21 \langle \{AD\}, \{A \rightarrow D\} \rangle, R22 \langle \{ABE\}, \{\underline{BE \rightarrow A}\} \rangle\}$ ; (依赖  $B \rightarrow D$  丢失)

R21 和 R22 均是 BCNF, 不需进一步分解。

因此, 将 R 分解为  $\rho = \{R1 \langle \{AC\}, \{A \rightarrow C\} \rangle, R21 \langle \{AD\}, \{A \rightarrow D\} \rangle, R22 \langle \{ABE\}, \{\underline{BE \rightarrow A}\} \rangle\}$

或  $\rho = \{R1 \langle \{AC\}, \{A \rightarrow C\} \rangle, R21 \langle \{BD\}, \{B \rightarrow D\} \rangle, R22 \langle \{ABE\}, \{\underline{BE \rightarrow A}\} \rangle\}$

5. 设有关系模式 R (U, F), 其中  $U = ABCDE$ ,  $F = \{A \rightarrow B, BC \rightarrow E, ED \rightarrow AB\}$ 。

(1) 计算  $(CD)^+_F$ 、 $(CDE)^+_F$ 、 $(ACD)^+_F$  及  $(BCD)^+_F$ ;

(2) 求 R 的所有候选键, 并说明理由;

(3) R 最高满足第几范式? 为什么?

(4) 若 R 不属于 BCNF, 试改进该关系数据库设计, 使它满足 BCNF。

解答:

(1)  $(CD)^+_F = CD$

$(BCD)^+_F = ABCDE$

$(ACD)^+_F = ABCDE$

$(CDE)^+_F = ABCDE$

(2) CD 为左部属性, 其属性集闭包不包含所有属性,  $(BCD)^+_F$ ,  $(ACD)^+_F$ ,  $(CDE)^+_F$  均包含所有属性, 所以 BCD、ACD、CDE 均是 R 的候选码。

(3) R 最高满足第 3 范式。因为 R 中的所有属性均为主属性, 没有任何非主属性对码

传递依赖或部分依赖。但存在主属性对码传递依赖或部分依赖，所以 R 不属于 BCNF。

(4)  $F = \{A \rightarrow B, BC \rightarrow E, ED \rightarrow AB\}$ ，由 Armstrong 公理还可推出  $AC \rightarrow E$ 。

考虑  $A \rightarrow B$ ，A 不是 R 的码，将 R 分解为

$\rho = \{R1 \langle \{AB\} \rangle, \{A \rightarrow B\} \rangle, R2 \langle \{ACDE\}, \{AC \rightarrow E, ED \rightarrow A\} \rangle\}$ ;

R1 是 BCNF，R2 不是，需进一步分解。

考虑  $ED \rightarrow A$ ，ED 不是 R2 的码，将 R2 分解为

$\rho = \{R21 \langle \{EDA\}, \{ED \rightarrow A\} \rangle, R22 \langle \{EDC\} \rangle\}$ ；R22 上没有非平凡的函数依赖。

或考虑  $AC \rightarrow E$ ，将 R2 分解为

$\rho = \{R21 \langle \{ACE\}, \{AC \rightarrow E\} \rangle, R22 \langle \{ACD\} \rangle\}$ ;

R21、R22 均是 BCNF，不需进一步分解。

因此，可将 R 分解为  $\rho = \{R1 \langle \{AB\}, \{A \rightarrow B\} \rangle, R21 \langle \{EDA\}, \{ED \rightarrow A\} \rangle,$

$R22 \langle \{EDC\} \rangle\}$  (可表示为  $\rho = \{AB, EDA, EDC\}$ ) ;或将 R 分解为  $\rho = \{R1 \langle \{AB\},$

$\{A \rightarrow B\} \rangle, R21 \langle \{ACE\}, \{AC \rightarrow E\} \rangle, R22 \langle \{ACD\} \rangle\}$  (可表示为  $\rho = \{AB, ACE, ACD\}$ )。

同理，也可将 R 分解为  $\rho = \{BCE, AB, ACD\}, \rho = \{EDAB, DEC\}$  等。

6. 假设为自学考试成绩管理设计了一个关系 R (S#, SN, C#, CN, G, U)，其属性的含义依次为考生号、姓名、课程号、课程名、分数和主考学校名称。规定每个学生学习一门课程只有一个分数；一个主考学校主管多门课程的考试，且一门课程只能属于一个主考学校管理；每名考生有惟一的考号，每门课程有惟一的课程号。

(1) 根据题目所描述的语义写出关系模式 R 的基本函数依赖集。

(2) 确定关系模式 R 的候选键。

(3) 说明关系模式 R 最高达到第几范式？为什么？

(4) 将 R 规范化为 3NF，并具有无损连接性和保持函数依赖特性。

解答：

(1) 根据题目所描述的语义写出关系模式 R 的基本函数依赖集。

答:  $F = \{ (S\#, C\#) \rightarrow G, C\# \rightarrow U, S\# \rightarrow SN, C\# \rightarrow CN \}$

(2) 可确定关系模式 R 的候选键为  $(S\#, C\#)$ 。

(3) 最高达到第一范式。因为  $(S\#, C\#)$  是主键, 但存在函数依赖  $C\# \rightarrow U, S\# \rightarrow SN, C\# \rightarrow CN$ , 即 U、SN、CN 这些非主属性均部分函数依赖于主键, 所以只能达到第一范式。

(4) 根据算法 5.4, 可将关系模式分解为

$\rho = \{ S(\{S\#, SN\}, \{S\# \rightarrow SN\}), C(\{C\#, CN, U\}, \{C\# \rightarrow U, C\# \rightarrow CN\}), SC(\{S\#, C\#, G\}, (S\#, C\#) \rightarrow G) \}$

#### 五、证明题

1. 在关系模式  $R(U, F)$  中,  $X \rightarrow A \in F$ , 求证:  $F$  与  $G = F - \{X \rightarrow A\}$  等价的充要条件是  $A \in X_G^+$

证明: 由引理 5.3 可知:  $F^+ = G^+$  的充分必要条件是  $F \subseteq G^+$  且  $G \subseteq F^+$ 。

因为  $G = F - \{X \rightarrow A\}$ , 所以  $G \subseteq F^+$

而要证  $F \subseteq G^+$ , 只要证  $X \rightarrow A \in G^+$ , 即  $A \in X_G^+$ 。

因此,  $F$  与  $G = F - \{X \rightarrow A\}$  等价的充要条件是  $A \in X_G^+$ 。

亦可按引理 5.3 的证明来证。

2. 在关系模式  $R(U, F)$  中,  $X \subseteq U$ ,  $Z$  是  $X$  的真子集, 求证:  $F$  与  $\{F - \{X \rightarrow A\}\} \cup \{Z \rightarrow A\}$  等价的充要条件是  $A \in Z_F^+$ 。

证明同上。

## 习题

### 一、填空题

1. 数据库系统中要存储数据描述、数据本身、数据之间的联系和存取路径 等 4 个方面的数据。
2. 有关数据的描述称为元数据，其存储在数据库系统的数据字典 中。
3. 存取路径是实现数据存储 和 访问 的基本手段，是访问一个关系表文件中行集合的特殊技术。
4. 数据库通常存储在大容量的外部存储设备上，磁盘 是目前常用的外部存储器。
5. 数据在磁盘上以称为块 的定长存储单位形式组织。
6. 若关系表文件记录中有一个名为照片的字段，其中拟存放位图（bmp 文件），则该字段的类型应为BLOB（二进制大对象）。
7. 如果一个文件是聚集 的，则这个文件中元组紧缩到能存储这些元组的尽可能少的块中。
8. 在数据库管理系统中，通过 SQL 提供的CREATE INDEX 语句命令建立索引文件。利用索引文件建立索引键值和记录地址 之间的映射，可基于索引键高效地存取记录。目前大多数数据库系统使用B-树 数据结构来实现动态多级索引。
9. 按关键字集高效检索文档的最简单、最常用的方法是采用倒排索引。
10. 使用位图索引的优势表现为可高效地实现部分匹配 查询和范围 查询。

### 二、选择题

1. 在数据库系统的三级模式结构中，定义索引的组织方式属于D。  
A. 概念模式      B. 外模式      C. 逻辑模式      D. 内模式
2. 数据字典中不包含以下哪一项D。  
A. 学生表的定义  
B. 安全性和完整性约束规则  
C. 计算机系学生视图的定义  
D. 学生“张三”的信息
3. 根据关系中某些属性值的排序顺序存储记录的文件称为B。  
A. 堆文件      B. 顺序文件      C. 聚集文件      D. 散列文件
4. 下述关于散列文件的说法正确的是A。  
A. 一个散列桶中存放散列函数值相同的多个记录  
B. 不同散列键值的记录不可能对应于同一个散列函数值  
C. 同一桶内记录的散列键值是相同的  
D. 散列键值必须为整型数

5. 下列关于索引的叙述, 正确的是 A。

- A. 可以根据需要在基本表上建立一个或多个索引, 从而提高系统的查询效率
- B. 一个基本表最多只能有一个索引
- C. 建立索引的目的是为了给数据表中的元素指定别名, 从而使别的表也可以引用这个元素
- D. 一个基本表上至少要存在一个索引

6. 下列关于索引的叙述, 正确的是 A C D。

- A. 数据文件最多只能在一个排序键上排序, 最多只有一个聚集索引
- B. 数据文件最多只有一个非聚集索引
- C. 稀疏索引只能建立在顺序文件上
- D. 稀疏索引一定是聚集的, 而非聚集索引是稠密的

### 三、简答题

1. 简述数据字典的内容和作用。
2. 为什么数据库系统中在主键上都默认创建索引?
3. 常见的文件组织有那些? 各有什么优缺点?
4. 试解释为什么一个文件只能有一个聚集索引。
5. 试解释为什么一个非聚集索引必须是稠密的。
6. 试解释为什么一个稀疏索引必须是聚集的。
7. 多级索引如何提高查找文件的效率。
8. 解释多级索引(树索引)和散列索引哪种适合用于等值搜索, 哪种适合范围搜索?
9. B<sup>+</sup>树的阶数 m 是什么? 描述 B<sup>+</sup>树的内部结点和叶结点的结构。
10. 在图 6-30 中的 B<sup>+</sup>树上执行以下操作, 描述那些引起树改变的操作所带来的变化。
  - (a) 查找键值为 41 的记录;
  - (b) 查找键值小于 30 的记录;
  - (c) 查找键值在 20 和 30 之间的所有记录;
  - (d) 插入键值为 1 的记录;
  - (e) 删除键值为 23 的记录;
  - (f) 删除键值从 14 到 16 的所有记录;
  - (g) 删除键值大于等于为 23 的所有记录;

### 四、计算题

1. 假设一条记录有如下顺序的字段: 一个长度为 23 的字符串, 一个 4 字节整数, 一个 SQL 日期(8 字节)。且记录有一个 12 字节的首部, 在下列几种情况下, 这条记录占用多少字节?

- (a) 字段可在任何字节处开始;

(b) 字段必须在 4 的倍数的字节处开始;

(c) 字段必须在 8 的倍数的字节处开始。

解答:

(a)  $12+23+4+8=47$  (字节)

(b)  $12+24+4+8=48$  (字节)

(c)  $16+24+8+8=56$  (字节)

2. 一个病人记录包含以下定长字段: 病人的出生日期 (8 字节), 身份证号码 (18 字节), 病人 ID (9 字节); 还包括变长字段: 姓名, 住址和病史。记录内一个指针需要 8 字节, 记录长度是一个 2 字节整数。假设不需要对字段进行对齐, 不包括变长字段空间, 这条记录需要多少字节?

解答:

3 个变长字段需要 2 个指针, 这条记录需要

$$8*2+8+18+9+2=53$$

3. 假设一个磁盘块可存放 5 个记录, 或 20 个键-指针对。已知一个数据库文件有  $n$  个记录, 如果表示成  $n$  的函数, 创建该数据文件的稠密索引和稀疏索引各需多少磁盘块? 如果使用多级索引, 并且最后一级的索引只能包含一个磁盘块, 又各需多少磁盘块?

解答:

创建该数据文件的稠密索引需  $(n/20)$  个磁盘块;

假设文件聚集存储, 创建该数据文件的稀疏索引需  $(n/5/20)$  个磁盘块;

如果使用多级索引, 建立一级稠密索引、二级稀疏索引、三级稀疏索引则需

$$n/20 + n/400 + n/8000 \text{ (磁盘块)}$$

当  $n$  很大时, 可能需建更高一级的索引, 直到  $n \leq 20^k$ ,  $k$  为索引级数。

如果使用多级索引, 建立一级稀疏索引、二级稀疏索引、三级稀疏索引则需

$$n/100 + n/2000 + n/40000 \text{ (磁盘块)}$$

当  $n$  很大时, 可能需建更高一级的索引, 直到  $n \leq 5*20^k$ ,  $k$  为索引级数。

4. 假定指针占 4 字节, 而键占 20 字节, 大小为 16384 字节的块可存放多少个键和指针?

解答:

假定在索引文件的 B+树组织中, 指针占 4 字节, 而键占 20 字节, 大小为 16384 字节的块可存放的指针数为  $m$ , 则

$$20(m-1) + 4m \leq 16384, \text{ 取 } m=683$$

存放 682 键和 683 个指针。

如果采取索引顺序文件, 则存放 682 个键和 682 个指针。

5. B+树中内结点和叶结点的键和指针的最小数目在下列情况下分别是多少?

(a) 每块可存放 11 个键和 12 个指针;

(b) 每块可存放 12 个键和 13 个指针。

解答：

(a)  $M=12$

内结点的指针数  $\lceil m/2 \rceil \leq n \leq m$ ，即  $6 \leq n \leq 12$ ，键值数  $5 \leq n \leq 11$ 。

叶结点的键值数  $\lfloor (m-1)/2 \rfloor \leq n \leq m-1$ ，即  $6 \leq n \leq 11$ 。指针数  $7 \leq n \leq 12$

(b)  $M=13$

内结点的指针数  $7 \leq n \leq 13$ ，键值数  $6 \leq n \leq 12$ 。

叶结点的键值数  $6 \leq n \leq 12$ ，指针数  $7 \leq n \leq 13$ ，。



## 习题

### 一、填空题

1. 在关系数据库系统中, 由于实现了查询优化, 用户只要提出 做什么, 不必指出 怎样做。
2. 查询处理分为查询编译和 查询执行 两大步骤, 而查询编译又可细分为 查询分析 与 处理、查询优化 和 代码生成 等步骤。
3. 在 RDBMS 中, 查询编译器将查询语句经过分析与检查后转换为某种内部格式, 并可用 关系代数 等价地表示。
4. 代数优化是由查询优化器将关系代数初始查询树转换成一个预期所需执行时间较小的 等价 的关系代数查询树, 目标是得到一个可被转换成最有效的物理查询计划的一个“优化”的逻辑查询计划。
5. 关系代数表达式 (查询树) 的优化就是指按照一定的规则, 改变关系代数表达式中操作的 顺序 和 组合, 将其转换为一个可以更高效执行的关系代数表达式。
6. 每一个操作符的 算法 的选择是将逻辑查询计划转变为物理查询计划过程中的一个必不可少的部分。
7. 在关系代数运算中, 笛卡尔积 和 连接 运算是最耗费时间和空间的。究竟应采用什么样的策略才能节省时间和空间, 这就是优化的准则。

### 二、选择题

1. 设  $E$  是关系代数表达式,  $F$  是选择条件表达式, 并且只涉及  $A_1, \dots, A_n$  属性, 则有 A
  - A.  $\sigma_F(\pi_{A_1, A_2, \dots, A_n}(E)) \equiv \pi_{A_1, A_2, \dots, A_n}(\sigma_F(E))$
  - B.  $\sigma_F(\pi_{A_1, A_2, \dots, A_n}(E)) \equiv \sigma_F(\pi_{A_1, A_2, A_n}(E))$
  - C.  $\sigma_F(\pi_{A_1, A_2, \dots, A_n}(E)) \equiv \pi_{A_1}(\sigma_F(E))$
  - D.  $\sigma_F(\pi_{A_1, A_2, \dots, A_n}(E)) \equiv \pi_{A_1, A_2, \dots, A_n}(\sigma_F(\pi_{A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m}(E)))$
2. 设  $E$  是关系代数表达式, 若  $F$  中有不属于  $A_1, \dots, A_n$  的属性  $B_1, \dots, B_m$ , 则有 D。
  - A.  $(\sigma_F(\pi_{A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m}(E))) \equiv \sigma_F(\pi_{A_1, A_2, \dots, A_n}(E))$
  - B.  $(\sigma_F(\pi_{A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m}(E))) \equiv \sigma_F(\pi_{A_1, A_2, A_3}(E))$
  - C.  $(\sigma_F(\pi_{A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m}(E))) \equiv \pi_{A_1}(\sigma_F(E))$
  - D.  $(\sigma_F(\pi_{A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m}(E))) \equiv \pi_{A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m}(\sigma_F(E))$
3. 如果条件  $F$  不仅涉及  $L$  中的属性, 而且还涉及不在  $L$  中的属性  $L_1$ , 则有 B。
  - A.  $\pi_L(\sigma_F(E)) \equiv \pi_L(\sigma_F(\pi_{L \wedge L_1}(E)))$
  - B.  $\pi_L(\sigma_F(E)) \equiv \pi_L(\sigma_F(\pi_{L \vee L_1}(E)))$
  - C.  $\pi_L(\sigma_F(E)) \equiv \sigma_F(\pi_{L \wedge L_1}(E))$
  - D.  $\pi_L(\sigma_F(E)) \equiv \sigma_F(\pi_{L \vee L_1}(E))$

4. 如果条件  $F$  形如  $F_1 \wedge F_2$ ,  $F_1$  仅涉及  $E_1$  中的属性,  $F_2$  涉及到  $E_1$  和  $E_2$  中的属性, 则有 D。

- A.  $\sigma_F(E_1 \times E_2) \equiv \sigma_{F_1}(E_1) \times \sigma_{F_2}(E_2)$
- B.  $\sigma_F(E_1 \times E_2) \equiv \sigma_{F_1}(\sigma_{F_2}(E_1) \times \sigma_{F_2}(E_2))$
- C.  $\sigma_F(E_1 \times E_2) \equiv \sigma_{F_2}(\sigma_{F_1}(E_1) \times \sigma_{F_2}(E_2))$
- D.  $\sigma_F(E_1 \times E_2) \equiv \sigma_{F_2}(\sigma_{F_1}(E_1) \times E_2)$

### 三、简答题

- 试述为什么要对关系代数表达式进行优化。
- 基于代数等价的启发式优化中应用的主要启发式规则有哪些?
- 对于学生-课程数据库, 查询计算机系学生选修的所有课程名称。若 SQL 查询如下:

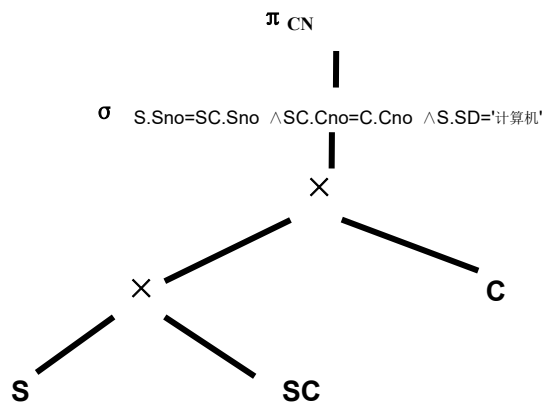
```
SELECT CN
FROM S, C, SC
WHERE S.Sno=SC.Sno AND
      SC.Cno=C.Cno AND
      S.SD='计算机';
```

要求画出其对应的初始关系代数查询树, 并用基于等价的关系代数启发式优化算法对其进行优化处理, 画出优化后的关系代数查询树。

解答:

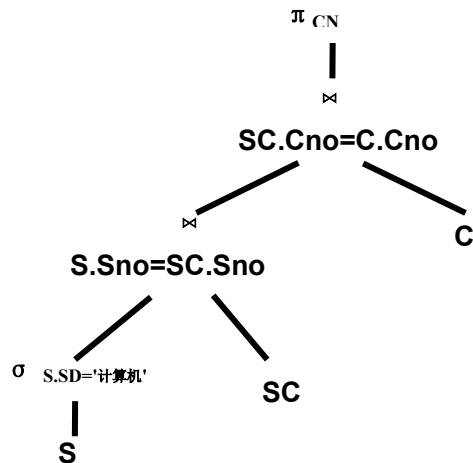
初始查询树的关系代数表达式及查询树

$$\pi_{CN} (\sigma_{S.Sno=SC.Sno \wedge SC.Cno=C.Cno \wedge S.SD='计算机'} (S \times SC \times C))$$



优化后的关系代数表达式及查询树

$$\pi_{CN} (((\sigma_{S.SD='计算机'} (S)) \bowtie SC) \bowtie C)$$



4. 一个查询的实际执行所需的代价包括哪些因素？哪些因素最常用作为代价估算的基础？

5. 选择操作的实现算法有哪些？其查询代价如何？

6. 连接操作的实现算法有哪些？其查询代价如何？

7. 以选择和连接操作为例，基于启发式规则进行物理优化的一些常用启发式规则有哪些？

#### 四、计算题

1. 假设  $B(R)=B(S)=10000$ ，并且  $M=1000$ 。计算嵌套循环连接的磁盘 I/O 代价。

解答：

为便于计算，建议将  $M$  值改为 1001。则基于块的嵌套循环连接的磁盘 I/O 代价（读数据）为：

$$B(S) + (B(S)B(R))/(M-1) = 10000 + 10000 \times 10000 / 1000 = 110000 (\text{次})$$

2. 假设  $B(R)=10000$ ， $T(R)=500000$ ， $R.a$  上有一个索引，令  $V(R,a)=k$ ， $k$  是某个常数。在下面的情况下，给出  $\sigma_{a=0}(R)$  的代价，表示为  $k$  的一个函数。忽略访问索引自身所需的磁盘 I/O。

(1) 索引是非聚集的；

(2) 索引是聚集的；

(3)  $R$  是聚集的，而且不使用索引。

解答：

$$B(R)=10000, T(R)=500000, V(R,a)=k,$$

则  $a=0$  的元组大约有  $T(R)/V(R,a)=500000/k$  个。可聚集存储在  $B(R)/V(R,a)=10000/k$  磁盘块上。则

(1) 若索引是非聚集的，则  $a=0$  的元组可随机分布在  $T(R)/V(R,a)$  磁盘块中，利用索引

估计要读取  $T(R)/V(R,a)$  次磁盘，即  $500000/k$  次。

(2) 索引是聚集的，则  $a=0$  的元组将顺序存储在连续的  $B(R)/V(R,a)$  磁盘块中，利用索引估计要读取  $\sigma_{a=0}(R)$  所需的磁盘 I/O 的次数将大约是  $B(R)/V(R,a)$ ，即  $10000/k$  次。

(3) 若  $R$  是聚集的，而且不使用索引，则  $a=0$  的元组可随机分布在  $B(R)$  中，读取  $\sigma_{a=0}(R)$  所需的磁盘 I/O 的次数将大约是  $B(R)$  次。

3. 考虑一个关系  $R(a,b,c,d)$ ，该关系有一个  $a$  上的聚集索引以及每一个其他属性上的非聚集索引。相关参数为： $T(R)=5000$ ， $B(R)=500$ ， $V(R,a)=50$ ， $V(R,b)=1000$ ， $V(R,c)=5000$ ， $V(R,d)=500$ 。给出下列选择运算的最佳查询计划及其磁盘 I/O 开销：

(1)  $\sigma_{a=1 \text{ AND } b=2 \text{ AND } c \geq 3}(R)$

(2)  $\sigma_{a=1 \text{ AND } b \leq 2 \text{ AND } c \geq 3}(R)$

(3)  $\sigma_{a=1 \text{ AND } b=2 \text{ AND } c=3}(R)$

解答：

$T(R)=5000$ ， $B(R)=500$ ， $V(R,a)=50$ ， $V(R,b)=1000$ ， $V(R,c)=5000$ ， $V(R,d)=500$ 。有一个  $a$  上的聚集索引以及每一个其他属性上的非聚集索引。则  $a=1$  的元组平均聚集存储在  $B(R)/V(R,a)=500/50=10$  个磁盘块中。则  $b=2$  的元组随机存储在  $T(R)/V(R,b)=5000/1000=5$  个磁盘块中。则  $c=3$  的元组随机存储在  $T(R)/V(R,c)=5000/5000=1$  个磁盘块中。

因存在索引，只考虑索引扫描方法，表扫描方法代价较高，不考虑。

(1)  $\sigma_{a=1 \text{ AND } b=2 \text{ AND } c \geq 3}(R)$

可能的查询计划的代价估算：

①使用  $a$  的索引进行索引扫描，找出  $a=1$  的元组，再从中过滤出  $b=2$  和  $c \geq 3$  的元组。则代价大约为 10 次磁盘 I/O。

②使用  $b$  的索引进行索引扫描，找出  $b=2$  的元组，再从中过滤出  $a=1$  和  $c \geq 3$  的元组。则代价大约 5 次磁盘 I/O。

③使用  $c$  的索引进行索引扫描，找出  $c \geq 3$  的元组，再从中过滤出  $a=1$  和  $b=2$  的元组。由于索引不是聚集的，因此需要大约  $T(R)/2=2500$  次磁盘 I/O。

因此，最佳查询计划是使用  $b$  的索引进行索引扫描，代价大约 5 次磁盘 I/O。

(2)  $\sigma_{a=1 \text{ AND } b \leq 2 \text{ AND } c \geq 3}(R)$

可能的查询计划的代价估算：

①使用  $a$  的索引进行索引扫描，找出  $a=1$  的元组，再从中过滤出  $b \leq 2$  和  $c \geq 3$  的元组。则代价大约为 10 次磁盘 I/O。

②使用  $b$  的索引进行索引扫描，找出  $b \leq 2$  的元组，再从中过滤出  $a=1$  和  $c \geq 3$  的元组。由于索引不是聚集的，因此需要大约  $T(R)/2=2500$  次磁盘 I/O。

③使用  $c$  的索引进行索引扫描，找出  $c \geq 3$  的元组，再从中过滤出  $a=1$  和  $b=2$  的元组。

由于索引不是聚集的，因此需要大约  $T(R)/2 = 2500$  次磁盘 I/O。

因此，最佳查询计划是使用 a 的索引进行索引扫描，代价大约 10 次磁盘 I/O。

(3)  $\sigma_{a=1 \text{ AND } b=2 \text{ AND } c=3}(R)$

①使用 a 的索引进行索引扫描，找出  $a=1$  的元组，再从中过滤出  $b=2$  和  $c=3$  的元组。则代价大约为 10 次磁盘 I/O。

②使用 b 的索引进行索引扫描，找出  $b=2$  的元组，再从中过滤出  $a=1$  和  $c=3$  的元组。则代价大约 5 次磁盘 I/O。

③使用 c 的索引进行索引扫描，找出  $c=3$  的元组，再从中过滤出  $a=1$  和  $b=2$  的元组。则代价大约 1 次磁盘 I/O。

因此，最佳查询计划是使用 c 的索引进行索引扫描，代价为 1 次磁盘 I/O。

## 习题

### 一、填空题

1. DBMS 的基本逻辑工作单元是事务，它是用户定义的一组 SQL 语句或整个程序。
2. 事务必须具有的四个性质的：原子性、一致性、隔离性和持久性，简称为事务的 ACID 特性。
3. 一个事务中对数据库的所有操作是一个不可分割的操作序列，要么都做，要么都不做，这个性质称为事务的原子性。
4. 一个事务的执行不能被其他事务干扰的特性，称为事务的隔离性。
5. 数据库系统可能发生的故障有：事务故障、系统故障和介质故障。
6. 多个事务并发操作会带来：丢失修改、脏读和不可重复读三类数据不一致现象。
7. 并发控制的主要方法是采用封锁技术。
8. 用于数据库恢复的基本技术有登记日志文件和数据转储。
9. 如果事务是一致的，多个事务并发执行的整体效果等同于某一次序下事务串行执行的效果，那么该并发调度将保持数据库的一致性状态。
10. 若事务 T 对数据对象 A 加了 S 锁，则其他事务只能对数据 A 再加S 锁，不能加X 锁，直到事务 T 释放 A 上的锁。

### 二、选择题

1. DBMS 并发控制和恢复的基本单位是C。  
A. 表            B. 命令            C. 事务            D. 程序
2. 对事务回滚的正确描述是A。  
A. 将该事务对数据库的修改进行恢复  
B. 将事务对数据库的更新写入硬盘  
C. 跳转到事务程序的开头重新执行  
D. 将事务中修改的变量值恢复到事务开始时的初值
3. 若系统在运行过程中，由于某种硬件故障，使存储在外存上的数据部分损失或全部损失，这种情况称为C。  
A. 事务故障      B. 系统故障      C. 介质故障      D. 运行故障
4. 系统故障会造成A。  
A. 内存数据丢失            B. 硬盘数据丢失  
C. 软盘数据丢失            D. 磁带数据丢失
5. 在对数据库进行恢复时，对已经 COMMIT 但更新未写入磁盘的事务执行A操作。  
A. REDO          B. UNDO          C. ABORT          D. ROLLBACK

6. 事务的并发执行会带来\_\_D\_\_数据不一致性问题。
- A. 丢失修改、不可重复读、读脏数据、死锁
  - B. 不可重复读、读脏数据、死锁
  - C. 丢失修改、读脏数据、死锁
  - D. 丢失修改、不可重复读、读脏数据
7. 在 DB 技术中，“脏数据”是指\_\_B\_\_。
- A. 未回滚的数据
  - B. 未提交的数据
  - C. 回滚的数据
  - D. 未提交随后又被撤销的数据
8. 数据库中的封锁机制是\_\_C\_\_的主要方法。
- A. 完整性
  - B. 安全性
  - C. 并发控制
  - D. 恢复
9. 若事务 T 对数据对象 A 加上 S 锁，则\_\_C\_\_。
- A. 事务 T 可以读 A 和修改 A，其他事务只能对 A 加 S 锁，不能加 X 锁。
  - B. 事务 T 可以读 A 但不能修改 A，其他事务能对 A 加 S 锁和 X 锁。
  - C. 事务 T 可以读 A 但不能修改 A，其他事务只能对 A 加 S 锁，不能加 X 锁。
  - D. 事务 T 可以读 A 和修改 A，其他事务能对 A 加 S 锁和 X 锁。
10. 若事务 T 对数据 R 已加 X 锁，则其他事务对数据 R\_\_D\_\_。
- A. 可以加 S 锁不能加 X 锁
  - B. 不能加 S 锁可以加 X 锁
  - C. 可以加 S 锁也可以加 X 锁
  - D. 不能加任何锁

### 三、简答题

1. 数据库系统运行中可能产生的各类故障会导致数据库出现哪些错误状态？
2. 针对不同的故障，试给出恢复的策略和方法。
3. 具有检查点的恢复技术有什么优点？试述使用检查点方法进行系统恢复的步骤。
4. 什么是可串行化调度？什么是冲突可串行化的调度？
5. 登记日志文件的原則有哪些？
6. 严格的两阶段封锁协议包含哪些内容？
7. 什么是死锁？预防死锁的方法有哪些？
8. 请简述如何用事务等待图来预防和检测死锁？
9. 请简述在多粒度封锁协议中引入意向锁的作用？
10. ANSI 标准提供几种隔离级别选项？各级别如何使用锁？
11. 设  $T_1T_2T_3$  是如下三个事务：

$T_1: A:=A+2; \quad T_2: A:=A*2; \quad T_3: A:=A^2$

设 A 的初值为 0。

- (1) 若三个事务允许并发执行，则有多少种可能的正确的结果，请分别列举出来；
- (2) 请给出一个可串行化的调度，并给出执行结果；

- (3) 请给出一个非可串行化的调度，并给出执行结果；
- (4) 若三个事务都遵守两阶段封锁协议，请给出一个不产生死锁的可串行化调度；
- (5) 若三个事务都遵守两阶段封锁协议，请给出一个产生死锁的调度。

解答：

(1) 若三个事务允许并发执行，当 A 的初值为 0，则有 6 种可能的正确的结果，结果值只有 4 种。

	1	2	3	4	5	6
执 行 序 列	T <sub>1</sub> : A:=A+2 T <sub>2</sub> : A:=A*2 T <sub>3</sub> : A:=A <sup>2</sup>	T <sub>1</sub> : A:=A+2 T <sub>3</sub> : A:=A <sup>2</sup> T <sub>2</sub> : A:=A*2	T <sub>2</sub> : A:=A*2 T <sub>1</sub> : A:=A+2 T <sub>3</sub> : A:=A <sup>2</sup>	T <sub>2</sub> : A:=A*2 T <sub>3</sub> : A:=A <sup>2</sup> T <sub>1</sub> : A:=A+2	T <sub>3</sub> : A:=A <sup>2</sup> T <sub>1</sub> : A:=A+2 T <sub>2</sub> : A:=A*2	T <sub>3</sub> : A:=A <sup>2</sup> T <sub>2</sub> : A:=A*2 T <sub>1</sub> : A:=A+2
结 果	16	8	4	2	4	2

(2) 事务的内部结构如下表所示。一个可串行化的调度，实际上就是一个串行调度。可是上面任何一种调度序列。

T <sub>1</sub>	T <sub>2</sub>	T <sub>3</sub>
<b>Read(A,t)</b> <b>t:=t+2</b> <b>Write(A,t)</b>	<b>Read(A,s)</b> <b>s:=s*2</b> <b>Write(A,s)</b>	<b>Read(A,r)</b> <b>r:=r*r</b> <b>Write(A,r)</b>

(3) 下表给出一个非可串行化的调度，初值为 0 时，执行结果为 2。虽然为某一串行调度的结果，但不是可串行化的调度。当初值不为 0 时，结果不正确。比如初值为 2 时，结果为 4，并不与任何一个串行调度的结果相同（可参照（1）自行计算串行结果）。

T <sub>1</sub>	T <sub>2</sub>	T <sub>3</sub>
<b>Read(A,t)</b>  <b>t:=t+2</b>   <b>Write(A,t)</b>	<b>Read(A,s)</b> <b>s:=s*2</b>   <b>Write(A,s)</b>	<b>Read(A,r)</b> <b>r:=r*r</b> <b>Write(A,r)</b>



**操作序列为：R1(A), R2(A), R3(A), W3(A), W2(A), W1(A)**

彼此冲突，无法冲突转换为一个冲突可串行化的调度，因此不是一个可串行化的调度。

(4) 若三个事务都遵守两阶段封锁协议，一个不产生死锁的可串行化调度应该是一个串行调度。

(5) 若三个事务都遵守两阶段封锁协议，(3) 中的调度就是一个产生死锁的调度。

**R1(A), R2(A), R3(A), W3(A), W2(A), W1(A)**

T <sub>1</sub>	T <sub>2</sub>	T <sub>3</sub>
SLOCK(A) Read(A,t)	SLOCK(A) Read(A,s)	SLOCK(A) Read(A,r)
		XLOCK(A) 等待 Write(A,r)
	XLOCK(A) 等待 Write(A,r)	
XLOCK(A) 等待 Write(A,r)		

12. 在下面的每个动作序列中，假设共享锁恰好在每个读动作前申请，而排他锁恰好在每个写动作前申请。此外，解锁恰好发生在事务执行的最后一个动作后。说明哪些动作被拒绝以及是否有死锁发生，并说明在动作执行过程中等待图怎样演变。如果存在死锁，选择一个事务并将其中止，说明动作序列将怎样继续下去。

(1) r<sub>1</sub>(A); r<sub>3</sub>(B); r<sub>2</sub>(C); w<sub>1</sub>(B); w<sub>3</sub>(C); w<sub>2</sub>(D);

(2) r<sub>1</sub>(A); r<sub>3</sub>(B); r<sub>2</sub>(C); w<sub>1</sub>(B); w<sub>3</sub>(C); w<sub>2</sub>(A);

(3) r<sub>1</sub>(A); r<sub>3</sub>(B); w<sub>1</sub>(C); w<sub>3</sub>(D) ; r<sub>2</sub>(C); w<sub>1</sub>(B); w<sub>4</sub>(D); w<sub>3</sub>(A);

(4) r<sub>1</sub>(A); r<sub>3</sub>(B); w<sub>1</sub>(C); r<sub>2</sub>(D) ; r<sub>4</sub>(E); w<sub>2</sub>(B); w<sub>3</sub>(C); w<sub>4</sub>(A); w<sub>1</sub>(D);

解答：

(1) r<sub>1</sub>(A); r<sub>3</sub>(B); r<sub>2</sub>(C); w<sub>1</sub>(B); w<sub>3</sub>(C); w<sub>2</sub>(D);

T <sub>1</sub>	T <sub>2</sub>	T <sub>3</sub>
SLOCK(A) R1(A)		SLOCK(B) R3(B)
	SLOCK(C) R2(C)	
XLOCK(B) 等待 T3 W1(B)		XLOCK(C) 等待 T2 W3(C)

	<b>XLOCK(D)</b> <b>W2(D)</b> <b>UNLOCK(C)</b> <b>UNLOCK(D)</b>	<b>XLOCK(C)</b> <b>W3(C)</b> <b>UNLOCK(B)</b> <b>UNLOCK(C)</b>
<b>XLOCK(B)</b> <b>W1(B)</b> <b>UNLOCK(A)</b> <b>UNLOCK(B)</b>		

$w_1(B)$ 、 $w_3(C)$ 动作被拒绝，等待图为  $T_1 \rightarrow T_3 \rightarrow T_2$ 。不存在死锁。

(2)  $r_1(A)$ ;  $r_3(B)$ ;  $r_2(C)$ ;  $w_1(B)$ ;  $w_3(C)$ ;  $w_2(A)$ ;

<b>T<sub>1</sub></b> <b>SLOCK(A)</b> <b>R1(A)</b>  <b>XLOCK(B)</b> 等待 T3 <b>W1(B)</b>	<b>T<sub>2</sub></b>  <b>SLOCK(C)</b> <b>R2(C)</b>  <b>XLOCK(A)</b> 等待 T1 <b>W2(A)</b>	<b>T<sub>3</sub></b>  <b>SLOCK(B)</b> <b>R3(B)</b>  <b>XLOCK(C)</b> 等待 T2 <b>W3(C)</b>
---	---	---

$w_1(B)$ 、 $w_3(C)$ 、 $w_2(A)$ 动作被拒绝，等待图为  $T_1 \rightarrow T_3 \rightarrow T_2 \rightarrow T_1$ 。存在死锁。  
 可将  $T_1$  中止，释放对 A 的锁， $w_2(A)$ 、 $w_3(C)$ 依次执行。

(3)  $r_1(A)$ ;  $r_3(B)$ ;  $w_1(C)$ ;  $w_3(D)$  ;  $r_2(C)$ ;  $w_1(B)$ ;  $w_4(D)$ ;  $w_3(A)$ ;

<b>T<sub>1</sub></b> <b>SLOCK(A)</b> <b>R1(A)</b>  <b>XLOCK(C)</b> <b>W1(C)</b>  <b>XLOCK(B)</b> 等待 T3 <b>W1(B)</b>	<b>T<sub>2</sub></b>  <b>SLOCK(C)</b> 等待 T1 <b>R2(C)</b>	<b>T<sub>3</sub></b>  <b>SLOCK(B)</b> <b>R3(B)</b>  <b>XLOCK(D)</b> <b>W3(D)</b>  <b>XLOCK(A)</b> 等待 T1 <b>W3(A)</b>	<b>T<sub>4</sub></b>   <b>XLOCK(D)</b> 等待 T3 <b>W4(D)</b>
--	--	--	--

$r_2(C)$ ;  $w_1(B)$ ;  $w_4(D)$ ;  $w_3(A)$ 动作被拒绝, 等待图为  $T_2 \rightarrow T_1 \rightarrow T_3 \rightarrow T_1$ ,  $T_4 \rightarrow T_3$ 。存在死锁。

将  $T_3$  中止, 释放对 B、D 的锁,  $w_1(B)$ ;  $w_4(D)$ ;  $r_2(C)$ 依次执行。

(4)  $r_1(A)$ ;  $r_3(B)$ ;  $w_1(C)$ ;  $r_2(D)$  ;  $r_4(E)$ ;  $w_2(B)$ ;  $w_3(C)$ ;  $w_4(A)$ ;  $w_1(D)$ ;

T <sub>1</sub>	T <sub>2</sub>	T <sub>3</sub>	T <sub>4</sub>
SLOCK(A) R1(A)		SLOCK(B) R3(B)	
XLOCK(C) W1(C)	SLOCK(D) R2(D)		SLOCK(E) R4(E)
	XLOCK(B) 等待 T3 W2(B)	XLOCK(C) 等待 T1 W3(C)	XLOCK(A) 等待 T1 W4(A)
XLOCK(D) 等待 T2 W1(D)			

$w_2(B)$ ;  $w_3(C)$ ;  $w_4(A)$ ;  $w_1(D)$ 动作被拒绝, 等待图为  $T_2 \rightarrow T_3 \rightarrow T_1 \rightarrow T_2$ ,  $T_4 \rightarrow T_1$ 。存在死锁。

将  $T_1$  中止, 释放对 A、C 的锁,  $w_3(C)$ ;  $w_4(A)$ ;  $w_2(B)$ 依次执行。

**需要说明:**

题中表格中的操作序列只是操作先后的一个示意, 并不是指操作动作的串行顺序。

## 习 题

### 一、填空题

1. 在数据库设计中，规划存储结构和存取方法属于物理设计。
2. 数据库设计的几个步骤是：需求分析、概念设计、逻辑设计、物理设计、数据库实施、数据库运行与维护。
3. 在数据库设计中，使用 E-R 图工具的阶段是概念设计设计阶段。
4. 在数据库设计中，索引的设计属于物理设计的任务。
5. E-R 模型的三要素为：实体、属性和联系。
6. E-R 模型中，描述“包含”关系的联系名为ISA。

## 二、选择题

1. 从 E-R 模型向关系模型转换时, 一个 M:N 联系转换为关系模式时, 该关系模式的关键字是 **C** 。

### A.M 端实体的关键字

### B.N 端实体的关键字

### C.M 端实体关键字与 N 端实体关键字组合

#### D.重新选取其他属性

2. 概念结构设计阶段得到的结果是 B。

### A. 数据字典描述的数据需求

### B. E-R 图表示的概念模型

C. 某个 DBMS 所支持的数据模型

D. 包括存储结构和存取方法的物理结构

3. 在关系数据库的设计中, 设计关系模式是 C 的任务。

### A. 需求分析阶段

## B. 概念设计阶段

### C. 逻辑设计阶段

### D. 物理设计阶段

4. 逻辑结构设计阶段得到的结果是 C。

### A. 数据字典描述的数据需求

### B. E-R 图表示的概念模型

C. 某个 DBMS 所支持的数据模型

D. 包括存储结构和存取方法的物理结构

5. 物理结构设计阶段得到的结果是 D 。

### A. 数据字典描述的数据需求

### B. E-R 图表示的概念模型

C. 某个 DBMS 所支持的数据模型

#### D. 包括存储结构和存取方法的物理结构

6. 在关系数据库的设计中, 设计视图是 C 的任务。

### A. 需求分析阶段

## B. 概念设计阶段

### C. 逻辑设计阶段

### D. 物理设计阶段

### 三、简答题

1. 试述数据库设计过程及其各阶段上的设计描述。

2. 数据库概念结构设计和逻辑结构设计需要做哪些工作？两种结构设计分别得到哪些结果信息？

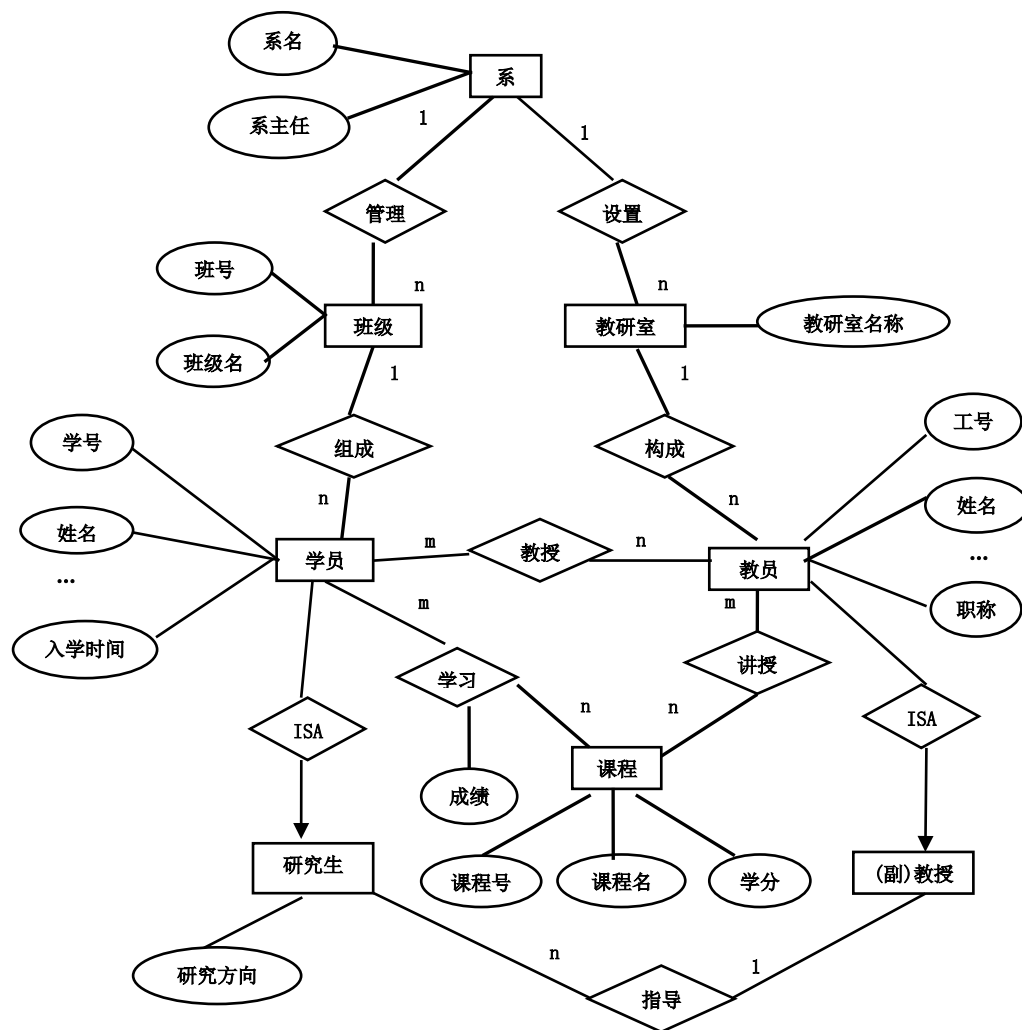
### 3. 规范化设计理论对数据库设计有什么指导意义?

4. 什么是数据库的重组和重构造？为什么要进行数据库的重组和重构造？如何进行数据库的重组和重构造？

5. 什么是 E-R 图？构成 E-R 图的基本要素是什么？

#### 四、设计题

1. 学校中有若干系, 每个系有若干个班级和教研室, 每个教研室有若干教师, 其中有的教授和副教授每人各带若干研究生, 每个班有若干学生, 每个学生选修若干门课, 每门课有若干名学生选修。用 E-R 图设计此学校的概念模型。



2. 假定一学生运动会管理系统的数据库包括以下的信息:

(1) 有若干班级, 每个班级的信息有: 班级号、班级名、专业、人数。

(2) 每个班级有若干运动员，每个运动员只能属于一个班，其信息有：运动员号、姓名、性别、年龄。

(3) 有若干比赛项目，其信息有：项目号、名称、比赛地点。

(4) 每名运动员可参加多项比赛，每个项目可有多人参加。

(5) 要求能记录每个比赛项目的运动员的名次与成绩。

要求：

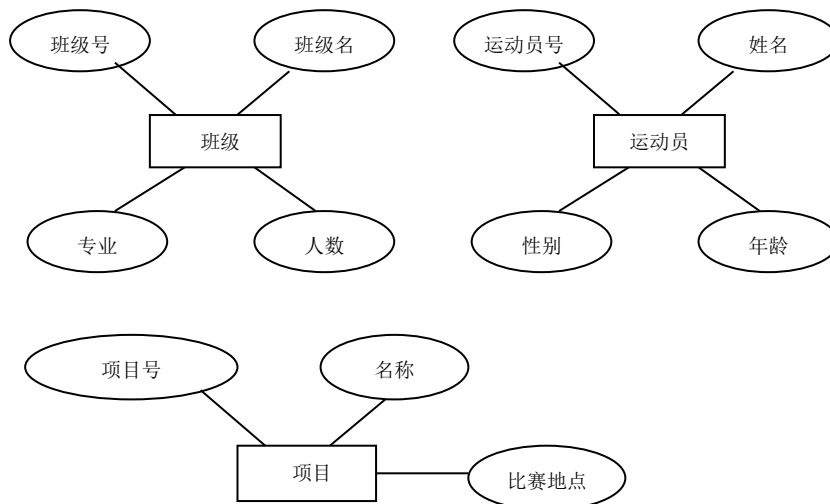
(1) 确定所涉及的实体及其包含的属性，并设计用 E-R 模型表示的概念设计模型；

(2) 根据该概念模型设计系统的关系数据库逻辑模型，指出各关系模式的主键和外键。；

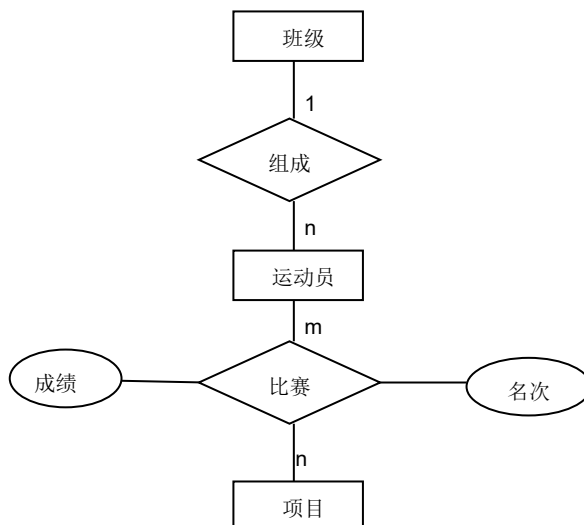
(3) 用 SQL 语言定义有关比赛信息的关系模式（要求包含相关的完整性约束）。

解答：

(1) 实体属性图：



E-R 概念模型：



说明：若将各分图合在一起也可以。

(2) 根据该概念模型设计系统的关系数据库逻辑模型:

**班级 (班级号, 班级名, 专业, 人数);** “班级号” 属性为该模式的主键。

**运动员 (运动员号, 姓名, 性别, 年龄);** “运动员号” 属性为该模式的主键。

**属于 (运动员号, 班级号);** “运动员号” 属性为该模式的主键, 也是该模式的外键, 对应 “运动员” 模式的主键 “运动员号”, “班级号” 为外键, 对应 “班级” 模式的主键 “班级号”。

**项目 (项目号、名称、比赛地点);** “项目号” 属性为该模式的主键。

**比赛 (项目号, 运动员号, 成绩, 名次);** “项目号, 运动员号” 属性组为该模式的主键, “项目号”, “运动员号” 为外键, 分别参照 “项目” 模式的主键 “项目号” 和 “运动员” 模式的主键 “运动员号”。

说明: 班级和运动员之间是一对多的关系, 所以也可将“属于”模式合并到“运动员”模式中, 将“班级号”作为运动员的属性。即

**运动员 (运动员号, 姓名, 性别, 年龄, 班级号);** “班级号” 为外键, 对应 “班级” 模式的主键 “班级号”。

(3) 比赛信息的关系模式定义

CREATE TABLE 比赛

(项目号 CHAR(10),

运动员号 CHAR(10),

成绩 TIME,

名次 CHAR(1),

PRIMARY KEY (项目号, 运动员号),

FOREIGN KEY (项目号) REFERENCES 项目 (项目号),

FOREIGN KEY (运动员号) REFERENCES 运动员 (运动员号));

3. 有一图书管理系统, 经需求分析得到如下信息:

(1) 图书信息包括: 书号、书名、作者、数量、出版社、单价、架号;

(2) 出版社信息包括: 出版社名称、地址、电话、邮编、信箱;

(3) 读者信息包括: 借书证号、姓名、单位;

(4) 一个出版社可以出版多种图书, 但每本书只能在一个出版社出版, 出版应有日期和责任编辑;

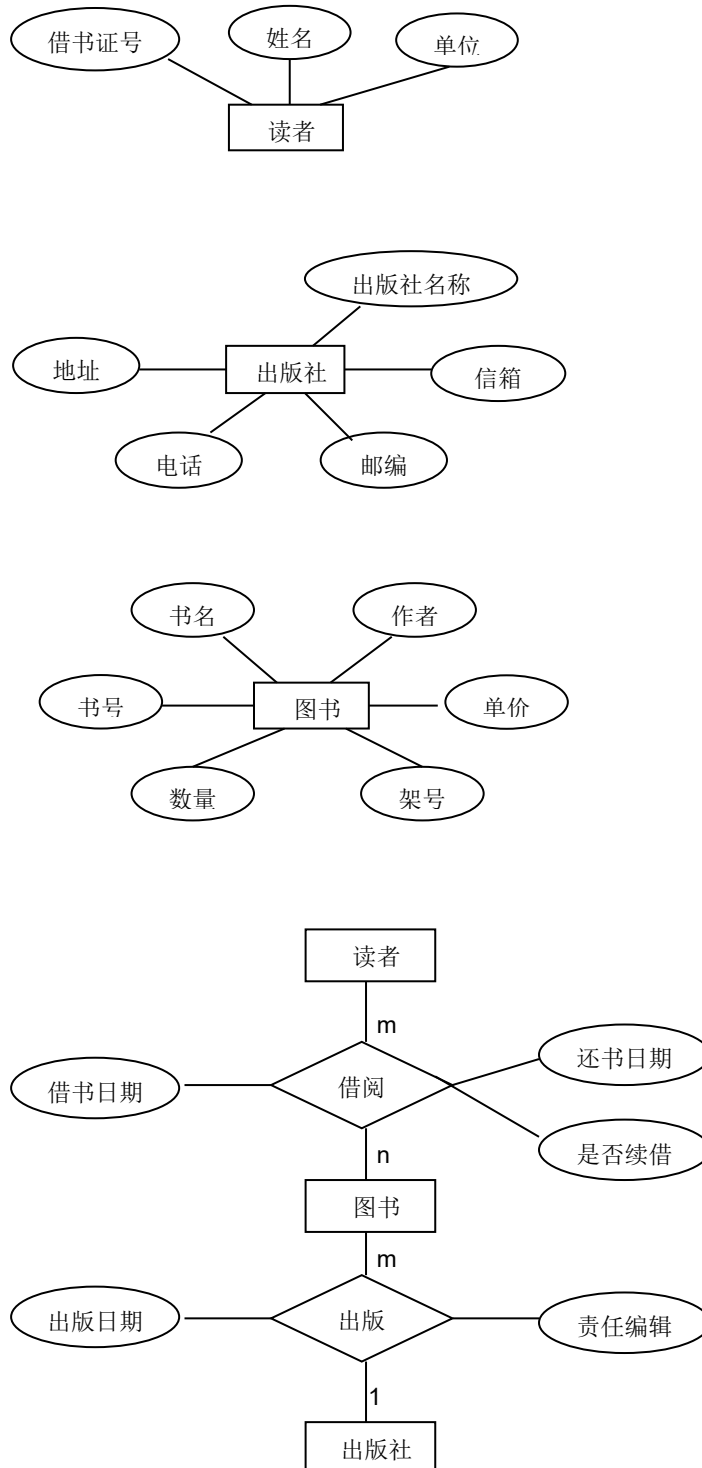
(5) 每个读者可以借阅多本图书, 每本图书可以有多人借阅。借阅信息包括: 借书日期、还书日期, 是否续借。

要求:

- (1) 确定所涉及的实体及其包含的属性，设计用 E-R 模型表示的概念模型；
- (2) 根据该概念模型设计系统的关系数据库逻辑模型，指出各关系模式的主键和外键。；
- (3) 用 SQL 语言定义有关图书出版信息的关系模式（要求包含相关的完整性约束）。

解答：

- (1) 各实体属性和它们之间联系的信息 E-R 模型如下



- (2) 该 E-R 模型转换为关系模式；

图书 (书号、书名、作者、数量、单价、架号)；“书号”为主键。



**出版社** (出版社名称、地址、电话、邮编、信箱); “出版社名称” 为主键。

**读者** (借书证号, 姓名, 单位); “借书证号” 为主键。

**借阅** (借书证号, 书号, 借书日期, 还书日期, 是否续借); “借书证号, 书号” 属性组为该模式的主键, “借书证号”, “书号” 为外键, 分别参照 “图书” 模式的主键 “书号” 和 “读者” 模式的主键 “借书证号”。

**出版** (书号, 出版社名称, 出版日期, 责任编辑); “书号” 属性组为该模式的主键, 也是该模式的外键, 对应 “图书” 模式的主键 “书号”, 出版社名称” 为外键, 对应 “出版社” 模式的主键 “出版社名称”。

说明: 出版社和图书之间是一对多的关系, 所以也可将 “出版” 模式合并到 “图书” 模式中, 即 “图书” 模式为:

**图书** (书号、书名、作者、数量、单价、架号、出版社名称、出版日期、责任编辑); “书号” 为主键。出版社名称” 为外键, 对应 “出版社” 模式的主键 “出版社名称”。

### (3) 图书出版信息的关系模式定义

CREATE TABLE 出版

(出版社名称 CHAR(40),

书号 CHAR(25),

出版日期 DATA,

责任编辑 CHAR(10),

PRIMARY KEY (书号),

FOREIGN KEY (出版社名称) REFERENCES 出版社(出版社名称));

### 4. 假定一个部门的数据库包括以下的信息:

职工的信息: 职工号、姓名、住址和所在部门。

部门的信息: 部门所有职工、经理和销售的产品及价格。

产品的信息: 产品名、制造商、价格、型号及产品内部编号。

制造商的信息: 制造商名称、地址、生产的产品名和型号。

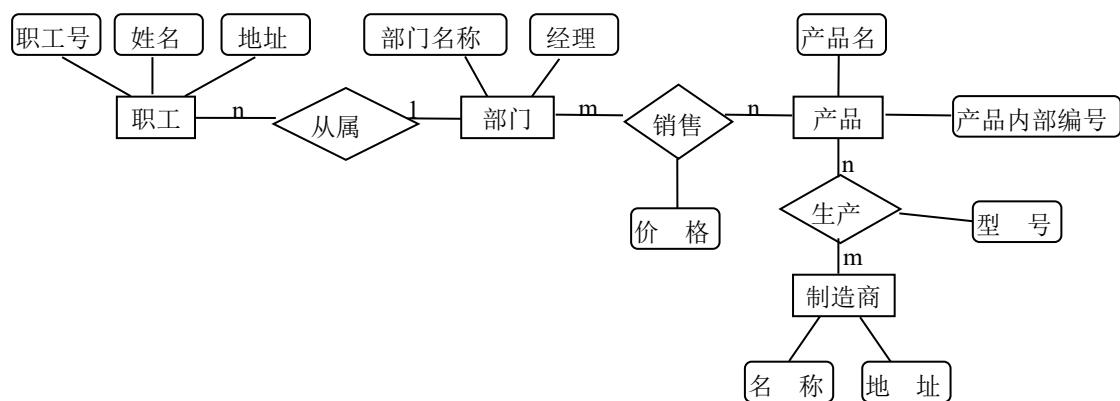
根据上述叙述, 解答以下问题:

(1) 试画出反映上述问题的 E-R 图;

(2) 将 E-R 模型转换成关系模型, 指出各关系模式的主键和外键。

解答:

(1) 用 E-R 模型表示的概念设计模型;



(2) 将 E-R 模型转换成关系模型：

职工（职工号，姓名，地址），“职工号”为主键。

部门（部门名称，经理），“部门名称”为主键。

产品（产品内部编号，产品名），“产品内部编号”为主键。

制造商（制造商名称，地址），“制造商名称”为主键。

从属（职工号，部门名称），“职工号”为主键，也是外键，对应“职工”模式的主键“职工号”，“部门名称”为外键，参照“部门”模式的主键“部门名称”。

销售（部门名称，产品内部编号，价格），“部门名称，产品内部编号”属性组为该模式的主键，“部门名称”，“产品内部编号”为外键，分别参照“部门”模式的主键“部门名称”和“产品”模式的主键“产品内部编号”。

生产（产品内部编号，制造商名称，型号），“产品内部编号，制造商名称”属性组为该模式的主键，“制造商名称”，“产品内部编号”为外键，分别参照“制造商”模式的主键“制造商名称”和“产品”模式的主键“产品内部编号”。

说明：部门和职工之间是一对多的关系，所以也可将“从属”模式合并到“职工”模式中，“部门名称”作为“职工”的属性。即

职工（职工号，姓名，地址，部门名称），“职工号”为主键，“部门名称”为外键，参照“部门”模式的主键“部门名称”。